NINTENDO 3DS

MONTAMOS UM TOP 10 COM OS GAMES MAIS ESPERADOS DO PORTÁTIL!

NINTENDO BLAST

www.nintendoblast.com.br

METROID™ Other M

EDIÇÃO Nº10 JUL/2010

BY NC ND

ÍNDICE

PERFIL **Samus Aran** 03

MARIO BITS **Mario & Dragon Quest** 05

BLAST FROM THE PAST **Série Metroid** 07

PRÉVIA **Arc Rise Fantasia (Wii)** 15

ANÁLISE **Dragon Quest IX (DS)** 19

TOP 10 **Promessas do Nintendo 3DS** 23

PRÉVIA **Metroid: Other M (Wii)** 35

DEVELOPERS **Masahiro Sakurai** 43

CUSTOM BLAST **Um Slogan para o N-Blast!** 45

ANÁLISE **Green Day: Rock Band** 47

GAMEDEV **Trabalhando com Sprites** 50

INTRODUÇÃO

"Metroid" é uma das franquias mais amadas da Nintendo. Após criar e misturar gêneros, a série ganha sua segunda edição no Wii, mas desta vez, não se trata de um "Prime", e sim de um título totalmente novo, que promete levar Samus - mais uma vez - a um novo patamar. Nesta edição você confere uma prévia do título, uma retrospectiva de todos os games da série, além de matérias sobre grandes títulos como Arc Rise Fantasia, Dragon Quest IX e outros. E não se esqueça do nosso Top 10 com as grandes promessas do Nintendo 3DS, o novo portátil revolucionário da Nintendo! Boa leitura!

- Sérgio Estrella

APOIAM A REVISTA

NINTENERDS

Meio Bit Games

Reino do Cogumelo

Game Reporter

Console Sonoro

Portallos

WarpZona

NDS Brasil

WiiClube

Nintendo World

DS Party

EGW

COMUNIDADE Nintendo Wii Brasil

revista digital ARKADE

REDAÇÃO

- Rafael Neves
- Alveni Lisboa
- Filipe Gatti
- Sérgio Estrella
- Sérgio Oliveira
- Gustavo Assumpção
- Eduardo Jardim
- Leonardo Marinho

REVISÃO

- Alveni Lisboa
- Bruna Lima
- Rafael Neves
- Sérgio Estrella
- Sérgio Oliveira
- Pedro Zambarda
- Douglas Fernandes

DIAGRAMAÇÃO

- Sérgio Estrella
- Gustavo Assumpção
- Felipe de França
- Rafael Esau
- Alex Silva
- Otávio Santos

Por **Filipe Gatti**

Perfil

Samus Aran

Após dez jogos, cerca de treze milhões de cópias vendidas e infinitos inimigos enviados para o além, podemos concluir de forma unânime: Samus está acostumada a ser só.

Samus Aran é a protagonista da série Metroid, criada e popularizada em 1986 no NES. Ela é considerada um dos maiores mascotes da Nintendo, principalmente por ser uma personagem do sexo feminino fazendo um papel que é dominado pelos homens nos videogames.

A história de Samus começa quando ela se tornou órfã, após seu planeta K-2L ter sido invadido e dizimado pelos conhecidos Space Pirates e seu líder, Ridley. Ela foi então adotada por uma raça alien chamada Chozo, que a levou para o planeta Zebes e a treinou para se tornar uma grande guerreira.

Após alcançar a fase adulta, Samus decidiu se juntar à Federação Policial, onde serviu até desertar sem nenhuma explicação. Anos mais tarde ela seria reconhecida como caçadora de recompensas.

De forma solitária, acredita-se que nesse tempo ausente Samus se dedicou unicamente à perseguição dos Space Pirates, procurando vingança por terem executado seus pais. Após ter criado fama e ser considerada a maior caçadora de recompensas do universo, ela foi procurada pela Federação Galáctica na missão de destruir todos os Metroids, organismos extremamente letais que estavam nas mãos dos Pirates. Até o presente momento, Samus continua recebendo missões da Federação e ainda persiste no objetivo de se vingar de Ridley, agindo sempre de forma solitária.

Samus teve seu sangue infundido com o DNA Chozo, o que lhe rendeu maior resistência aos ambientes desconhecidos, e também a possibilidade de utilizar a Power Suit, armadura que se alinha biologicamente com o corpo e mente da heroína. Sua armadura é a responsável por permitir que sua força aumente inúmeras vezes, além de poder acoplar um imenso arsenal de armas e equipamentos que a tornam uma máquina mortífera única.

A personagem já participou de dezenas de jogos da Nintendo, e também em outras mídias como a versão em quadrinhos de Captain N: The Game Master.

Lista de jogos em que Samus Aran é protagonista (ordem cronológica):

- Metroid – NES (1986);
- Metroid II: Return of Samus – Game Boy (1991);
- Super Metroid (1994); Metroid: Fusion – GBA (2002);
- Metroid Prime - GameCube (2002);
- Metroid Prime 2: Echoes – GameCube (2004);
- Metroid Zero Mission - GBA (2004);
- Metroid Hunters – NDS (2004);
- Metroid Prime 3: Corruption – Wii (2007);
- Metroid: Other M – Wii (2010).

Mario Bits

por Eduardo Jardim
www.reinodocogumelo.com

Mario, Dragon Quest e Banco Imobiliário

Hey, paisano! Bem-vindo a mais um Mario Bits, sua conexão entre o Mundo Real e o Marioverso aqui na Revista Nintendo Blast. Falando na revista, a anterior estava simplesmente saborosa, recheada com *Super Mario Galaxy 2*, E3 2010 e *Dragon Quest IX*, o mais novo título da série de RPG da Square Enix que promete encher o seu DS de diversão, estratégia e Slimes. Falando em DS, Square Enix e Mario, você já ouviu falar do crossover dos personagens do Mundo do Cogumelo com a turma de Dragon Quest? Se a resposta for não, amigo, estou aqui!

O game se chama *Itadaki Street DS*, criado pela mesma desenvolvedora por trás de *Final Fantasy* e *Super Mario RPG: Legend of the Seven Stars* (o último, quando a empresa ainda era conhecida como Squaresoft). Falando mais amplamente, a série Itadaki Street é uma coleção de jogos de tabuleiro criados pelo designer de Dragon Quest, Yuji Horii. O primeiro jogo foi lançado para Famicom em 1991; na época, ninguém sabia que o game se tornaria uma série de 7 jogos que se adaptavam às plataformas conforme a evolução dos video games. Começando pelo Famicom, seguindo para o Super Famicom, desviando-se da Nintendo ganhando portes para o PlayStation, PlayStation 2 e PSP, o filho pródigo retornou à casa N com tanta força que arrastou até mesmo os personagens do Mario para dentro do tabuleiro.

Lançado no dia 21 de junho somente no Japão, Itadaki Street DS continua seguindo o legado da série com uma jogabilidade similar ao jogo *Monopoly* – um nome bonito para o *Banco Imobiliário* que conhecemos. Como Mario e amigos já haviam se tornado superastros de jogos de tabuleiro, Itadaki Street DS ganhou elementos de *Mario Party*, podendo ser considerado uma mistureba entre as duas modalidades.

No game, os jogadores se movem por um tabuleiro, comprando lojas e coletando dinheiro de outros que dão as caras em seus pontos de venda. Dependendo das circunstâncias, os jogadores podem dominar o espaço de outros vendedores ou simplesmente torná-los inativos durante um turno. Somado a isso, os jogadores podem adquirir espaços exclusivos,

nos quais os demais participantes devem pagá-lo uma certa quantia ao pararem. Em Itadaki Street DS você também encontra as mecânicas de bônus e de atualização de nível - mais conhecido como level-up; eles podem ser obtidos passando por 4 espaços especiais no tabuleiro, indicados pelos naipes Paus, Ouros, Copas e Espadas. Tais naipes também podem afetar a jogabilidade ou os jogadores caso eles queiram arriscar um "cartão da sorte". O objetivo? Atingir um determinado valor de propriedade ou fazer um amigo falir. Vale lembrar também do "Casino", um espaço onde jogos temáticos podem acontecer.

Há 22 personagens jogáveis ao todo, 11 de cada série. De um lado, estão os personagens do encanador, que consistem de Mario, Luigi, Yoshi, Peach, Donkey Kong, Wario, Daisy, Bowser, Toad... e até mesmo Waluigi e Birdo, que mesmo sendo menores, não escaparam do tabuleiro. Os membros da turma do Dragon Quest incluem o Slime, Angelo, Jessica, Pudding, Bianca, Dragonlord, Alena, Kiryl, Hassan, Mommonja e a versão mirim de Yangus. Sem contar que você também pode dar as caras com Lakitus, Heal Slimes e Dancing Jewels (ou GoodyBags), que interferem e afetam a jogabilidade. As fases do jogo também são matematicamente divididas entre as séries: há 7 para cada um, que vão de lugares como *Mario Circuit* (de Mario Kart), *Yoshi's Island* e a Ilha Delfino (de *Super Mario Sunshine*) até as mais tenebrosas de Dragon Quest, como o Vulcão da Morte, o Navio Fantasma e o Labirinto de Mommonja.

Não pense que este jogo não tem mais nada de especial: na fase tutorial, você será lançado ao passado, de volta aos velhos tempos de Super NES, com a canção "*Hello, Happy Kingdom*", música tema do Reino do Cogumelo em Super Mario RPG. E além disso, este é um dos pouquíssimos jogos onde Mario possui um diálogo escrito, algo massivamente raro hoje em dia. Nós recomendamos que você dedique um tempo para se aventurar por Dragon Quest, se divertir com Itadaki Street e continuar ligado nas notícias e curiosidades do mundo dos video games.

por **Rafael Neves**

Dos primeiros pixels até texturas detalhadíssimas, a série Metroid passou vários anos se desenvolvendo em quase todas as plataformas Nintendo.As aventuras da caçadora Samus em busca de vingança e destruição total dos Metroids através das estrelas começaram quando o NES estourou e agora está prestes a ressurgir no Wii com Metroid: Other M. Basta compararmos um sprite de Samus em sua gênese com uma imagem em alta qualidade da mesma nos dias atuais para termos noção das mudanças sofridas pela série. Mas como se deu essa mudança? Todas as dúvidas serão esclarecidas nesse Blast from the Past de uma das maiores séries de vídeo-games do mundo: Metroid. Vamos relembrar cada um dos jogos da série principal, colocando de lado os spin-offs e remakes!

O INÍCIO

Samus Aran era apenas uma garotinha de 3 anos que vivia na colônia terrestre de K-2L quando um ataque dos Space Pirates destruiu para sempre a paz e a tranquilidade de sua vida. Rodney Aran e Virginia Aran eram seus pais, ambos trabalhavam numa nova forma de energia capaz de abastecer a nave da Galact Federation. Ridley, general dos Space Pirates, invade a colônia para extinguir as pesquisas dos pais de Samus e anular as forças da federação. Uma grande batalha cai sobre K-2L, na qual quase todos são mortos. Ridley foge da colônia, mas ele não era o único sobrevivente. Em meio aos destroços da luta, os Chozo, alienígenas homem-pássaro com grande desenvolvimento científico, encontram Samus. Eles a levam para o seu planeta, Zebes, mas para isso modificam geneticamente a garota para ela conseguir sobreviver à atmosfera do planeta. Após anos de treinamento, Samus recebe a Power Suit dos Chozo, uma armadura que imita o esqueleto deles e é capaz de absorver outras armas, adicionando-as ao arsenal. Samus então se torna uma Bounty Hunter, caçadora de recompensas, completando missões consideradas impossíveis. Nesse meio tempo, o planeta Zebes é invadido pelos Space Pirates para servir de base de clonagem de uma forma de vida capaz de sugar a energia de seu hospedeiro até a morte. Eram os Metroids. Samus então é enviada para Zebes a fim de parar a clonagem, descobrindo que os Chozo foram extintos pelos piratas. Chegando lá, começa o primeiro jogo da série... É claro que essas informações são muito omissas no jogo, tendo de recorrer ao mangá oficial para ter acesso.

A série Metroid foi originalmente idealizada em inspiração ao filme Alien e suas sequências. Não é a toa que o nome do pterodátilo Ridley vem do diretor Ridley Scott. Algumas cenas dos jogos também foram inspiradas nos filmes, como a cena final de Metroid II, onde Samus (uma protagonista feminina, assim como no filme) enfrenta a rainha dos Metroids após exterminar todos os outros (o mesmo ocorre em Alien, só que com os Aliens).

METROID
(NES E FAMICOM DISK SYSTEM)

Talvez um jogador da atualidade olhe para Metroid e veja-o como uma porcaria obsoleta, mas é o gamer retrô que enxerga a maravilha por trás deste cartucho. Metroid estreou no NES de uma forma não muito gritante, mas foi o começo de uma franquia muito preciosa para a Big N. Produzido por Gunpei Yokoi, dirigido por Yoshio Sakamoto e com Hirokazu Tanaka compondo as músicas, Metroid foi lançado para o Famicom Disk System em 6 de Agosto de 1986. Aqui nas Américas, ele desembarcou no NES no mesmo mês, mas um ano depois. O jogo apresentou uma fórmula que mesclava o side-scrolling de Super Mario Bros. com a mecânica de exploração de The Legend of Zelda. O híbrido dos dois sucessos iniciais da Nintendo no NES apresentava um cenário obscuro e futurista, no qual o jogador assumia o papel de Samus numa jornada solitária. A missão era invadir a base operacional dos Space Pirates no planeta Zebes e parar a clonagem dos Metroids, bem como garantir a destruição da base e de seus líderes: Ridley e Mother Brain. O jogo era bem simples, mas conseguiu criar um mapa extenso e com uma aparência realmente alienígena, mesmo com as dificuldades técnicas. O ritmo de progressão do game era bem parecido com o de Zelda, pois baseava-se nos itens acumulados. No fim da jornada, a revelação da dama por trás da armadura deixava os jogadores boquiabertos. Metroid foi considerado um dos melhores jogos já lançados para o NES, sendo refeito anos depois para o Game Boy Advance e vendido pelo Virtual Console do Wii.

METROID II: RETURN OF SAMUS
(GAMEBOY)

Somente em 1991 (Japão) e 1992 (América) que Metroid recebeu sua sequência. Zelda e Mario já haviam sido continuados há anos, mas Metroid permaneceu no esquecimento por muito tempo. A estréia da caçadora de recompensas em sua Power Suit no Game Boy com Metroid II: Return of Samus foi tão boa quanto no NES. Apesar do portátil da Nintendo não reproduzir cores, Metroid conseguiu criar outro mundo extenso e obscuro, trazendo mais um sucesso para o Game Boy. Mesmo sendo bem recebido na crítica e nas vendas e ter trazido grandes novidades, como os Save Points e itens exclusivos, Metroid II foi mal visto em alguns aspectos. Sobretudo na progressão do jogo, que não é mais determinada sutilmente pelos itens adquiridos, mas sim pelo número de Metroids mortos, aumentando a linearidade do game. Metroid II narrava a história de Samus após o primeiro game, mas hoje a série Prime preenche o vazio entre os dois games. Nesta sequência, a caçadora recebe a missão de ir até o planeta SR388, planeta natal dos Metroids, e destruir todos os espécimes vivos, garantindo a total erradicação dos Metroids. O jogo então vai avançando conforme você mata Metroids, inclusive há um contador no canto da tela. Ao fim do game, Samus leva consigo um bebê Metroid que a segue como uma mãe.

Metroid II: Return of Samus é uma exceção em muitos aspectos da série, sendo considerado um jogo esquecido hoje em dia. Ele foi o único a nunca ser lançado em outra plataforma, nenhuma de suas músicas foi rearranjada no futuro, é o único a não ter nenhum Space Pirate. Também é um dos poucos a não ter a aparição de Ridley.

SUPER METROID
(SUPER NINTENDO)

Demorou mais um tempo, mas enfim Metroid recebeu sua versão definitiva. Super Metroid foi um verdadeiro marco na história da franquia. O game para Super Nintendo (ou Super Famicom) melhorou os aspectos novos de Metroid II e trouxe uma característica chave do primeiro game, que foi perdida na sequência: as cores. Juntando tudo, Super Metroid conseguiu gerar um mundo altamente explorável e muito bem detalhado. O cenário é rico, os inimigos monstruosos e os efeitos de luz muito mais bonitos. Era possível realmente sentir que se estava num planeta alienígena e que as criaturas ao redor eram realmente de outro mundo. O jogo voltou ao ritmo de progressão original, baseado nos novos equipamentos e armas da Bounty Hunter. A jornada de Samus agora se tornou mais extensa e complexa, incluindo até um pouco de sentimento e confissões da caçadora.

Super Metroid chegou pertíssimo de ser cancelado três vezes durante o desenvolvimento, afinal, era um projeto muito grande se comparado aos últimos dois jogos. Isso sem falar que por pouco não saiu como bônus extra em Metroid Prime 2: Echoes.

Na cronologia, situa-se exatamente depois de Metroid II, quando Samus deixa o bebê Metroid numa base espacial de pesquisa. Lá, o último Metroid vivo é roubado por Ridley, o rival de Samus ressurge após ter sido reduzido a pó no primeiro Metroid. Samus segue o vilão até o planeta Zebes, onde descobre que a base dos Space Pirates foi reconstruída e está pronta para reiniciar a clonagem de Metroids. Não é preciso ser um gênio para saber que sua missão será destruir tudo lá dentro. Super Metroid tornou-se sucesso absoluto no Super Nintendo, mesmo que as vendas japonesas não tenham sido tão grandes quanto as americanas. No Japão ele chegou em 19 de Março de 1994, enquanto aqui no ocidente o lançamento se deu em 18 de Abril do mesmo ano. A cena final do game tornou-se símbolo da série Metroid. Com certeza esta é a versão definitiva de um Metroid bidimensional.

METROID FUSION
(GAMEBOY ADVANCE)

O primeiro Metroid para o Game Boy Advance marcou muito os fãs da série. Não falo exatamente pelo jogo em si, mas por que ele representou uma nova tomada de esperança, uma vez que a série Metroid caiu no esquecimento por oito anos. Durante a época do Nintendo 64, Samus só apareceu no Super Smash Bros., mas não recebeu nenhuma missão nova durante muito tempo. Eis que, com o novo portátil da Nintendo, abrem-se as portas para mais uma aventura da caçadora, lançada no dia 18 de Novembro de 2002 nas Américas e 14 de Fevereiro de 2004 no Japão. Metroid Fusion foi lançado junto à Metroid Prime (o qual falaremos daqui a pouco), se tornando um sucesso do GBA, com mais de 1 milhão de cópias vendidas no Japão e na América. Metroid Fusion trouxe melhorias para a série, tanto no visual mais trabalhado e vívido quanto nas novas habilidades de Samus. A começar pela armadura (Fusion Suit), que foge do estilo tradicional, terminando em upgrades exclusivos, como o Difusion Missile e a capacidade de se pendurar nas extremidades de elementos de cenário. No entanto, a linearidade do jogo foi um dos pontos negativos, já que o computador Adam dava todas as instruções ao jogador para prosseguir no game. Mesmo assim, o enredo mais profundo e repleto de revelações marcaram Metroid Fusion, que foi o primeiro jogo da série no qual Samus abriu a boca para proferir algo que não fossem gemidos de dor. A história começa anos após Super Metroid e também depois de Metroid: Other M. Samus retorna ao planeta SR388 acompanhando uma expedição científica, mas uma tragédia acontece quando Samus é infectada pelo parasita X. A infecção força os médicos a injetarem o DNA retirado do bebê Metroid de Super Metroid em Samus como única forma de salvar a caçadora, já que a espécie é predadora do X. A operação salva Samus por pouco, mas ela quase não tem tempo de descanso, pois parte

Segundo o manual de instruções de Metroid Fusion, os Metroids foram criados em laboratório pelos próprios Chozo para que estes erradicassem os parasitas X do planeta SR388. Com a extinção dos Metroids em SR388 em Metroid II, eles se multiplicaram sem parar e causaram o acidente de Metroid Fusion.

para a base espacial BSL, a fim de investigar a invasão do X por lá. BSL logo torna-se um lugar hostil, uma vez que o X consegue clonar toda a criatura já uma vez infectada, assim, o parasita consegue o DNA de vários monstros que estavam na base, incluindo Ridley. O final é mais uma vez surpreendente, contando com uma batalha contra o clone de Samus (gerado pelo X): SA-X e a revelação de uma conspiração para clonagem de Metroids em segredo.

Você sabia que, originalmente, Metroid Prime seria um jogo em 3D com visão em terceira pessoa? Há imagens e vídeos na internet do jogo em produção que mostram como ficaria Metroid Prime se assim fosse.

METROID PRIME
(GAME CUBE)

Metroid Prime foi outro marco na história da franquia Metroid. Até então, a série usava uma mecânica de movimentação side-scrolling em cenários bidimensionais. No entanto, eis que a Nintendo e a Retro Studios apresentam uma nova maneira de se jogar Metroid: como um FPS. De início, a comunidade gamer ficou em total alarde, afinal, haviam esperado oito anos por um jogo que fugia à tradicionalidade de um Metroid. Bom, Metroid Prime chegou ao Game Cube no dia 18 de Novembro de 2002 aqui no ocidente e 28 de Fevereiro de 2003 no Japão e galgou índices altíssimos nas críticas, acumulou prêmios e alcançou um bom número de vendas. Metroid é considerado não um mero First-Person-Shooter, mas sim um First-Person-Adventure, pois conta com uma mecânica de exploração nunca antes vista num FPS. Graficamente, Metroid Prime é excelente, pondo inveja até em jogos da geração atual. O planeta Tallon IV foi reproduzido com tamanha riqueza de detalhes, cada parte deste mundo é completamente diferente e muito bem feita. A mecânica em primeira pessoa funcionou muito bem, bem como as músicas muito bem trabalhadas. Novas habilidades como os diversos visores de Samus e a intensidade dos tiroteios em primeira pessoa agradaram muito. Cronologicamente, situa-se depois do primeiro Metroid, sendo o primeiro jogo da série a não ser um sucessor cronológico do antecessor. A história começava com Samus numa base espacial abandonada, onde ela encontra com seu rival Meta Ridley, agora com um corpo de metal. O pterodáctilo foge da luta para o planeta Tallon IV, Samus segue a criatura, mesmo tendo perdido diversas armas e funcionalidades da armadura durante a invasão à base espacial. Chegando em Tallon IV, Samus descobre mais sobre o povo Chozo e do grave problema que o planeta enfrenta com a substância radioativa: Phazon. O Phazon torna-se o foco do jogo, deixando a clonagem de Metroids um pouco à parte. Samus então descobre que os Space Pirates estão no planeta fazendo pesquisas a cerca do Phazon e sua capacidade mutagênica para criar outra arma biológica. O desenlace do jogo é, mais uma vez, épico. Uma das aventuras de Samus mais bem feita de todas.

METROID PRIME 2: ECHOES
(GAME CUBE)

Com o sucesso do primeiro game, a Retro Studios logo aproveitou a mecânica de Metroid Prime para desenvolver sua sequência: Echoes. O jogo usa o mesmo estilo de FPS do primeiro Prime; na verdade, é quase como uma expansão do antecessor. O jogo trouxe melhorias, aproveitando melhor alguns puzzles com a Morph Ball e a Spider Ball e acrescentou o Screw Attack à perspectiva 3D. O nível de dificuldade foi elevado e o primeiro modo multiplayer de um Metroid apareceu. A jogatina em grupo funcionava como qualquer outro FPS, não era algo espetacular, mas também não era chato nem enfadonho. Metroid Prime 2: Echoes também trouxe uma mecânica de antagonismo entre o mundo normal e o Dark. Essa funcionalidade trouxe novas armas, armaduras e um novo vilão: Dark Samus. Em Echoes, Samus parte para o planeta Aether a fim de investigar a perda de contato com uma equipe da Galact Federation. Lá, Samus descobre que o planeta foi divido em dois com o impacto de um cometa repleto de Phazon, criando uma dimensão alternativa: Dark Aether. Os Luminoth, povo que vivia no planeta, entraram em diversos conflitos com os Ing, raça nativa de Dark Aether e responsável pela destruição da tropa da Galact Federation. Nessa disputa, os Luminoth foram extintos, restando apenas U-Mos, o único sobrevivente. U-Mos encontra Samus e conta toda a história à Samus, implorando para que ela aceite ajudá-lo. Samus aceita e daí em diante uma jornada através de Aether e Dark Aether começa, contando a participação dos Space Pirates também. O sucesso não foi tão esmagador quanto o do primeiro, mas muito satisfatório. Lançado em 15 de Novembro de 2004 nas Américas e apenas 26 de Maio de 2005 no Japão.

Ridley é o vilão de Metroid que mais ressurge nos jogos da série, mas um dos games dos quais ele ficou de fora foi Echoes. O pior foi que ele deveria fazer uma aparição no jogo em sua forma Dark, inclusive há uma artwork mostrando como seria, mas acabou que Ridley não teve chance em Metroid Prime 2.

METROID PRIME HUNTERS
(NINTENDO DS)

Um dos primeiros jogos do novo portátil da Nintendo foi Metroid Prime Hunters, inclusive a demo First Hunt vinha junto a alguns modelos de Nintendo DS. Em 20 de Março de 2006 (para os americanos) e 1º de Julho de 2006 (para os japoneses), Metroid Prime Hunters chegou às lojas. Vendo pelas imagens, parece muito com um Prime qualquer, mas não era apenas isso. Hunters usava ao máximo o potencial do DS para criar gráficos lindos e um sistema multiplayer online e local bem confortáveis e divertidos. É claro que o modo single-player estava lá, narrando a aventura de Samus numa missão dentro do sistema Alimbic, disputando com outros caçadores de recompensas o segredo do povo que lá viveu. Mas o foco estava com certeza no modo para múltiplos jogadores. Nele, você pode escolher um dentre os vários Hunters do jogo para disputar contra outros jogadores. O charme do jogo estava em sua mecânica FPS pela touch screen do DS. A jogabilidade com a stylus garantiu uma experiência nova para os jogadores, pois dispensava o uso de dois analógicos (um para se locomover e outro para mirar) como nos FPS da época.

Na finalização 100% de Corruption, uma nave semelhante à do pirata Sylux de Metroid Prime Hunters segue Samus pelo espaço. Até hoje uma explicação para isso nunca apareceu.

METROID PRIME 3: CORRUPTION
(NINTENDO WII)

Passou-se a geração do Game Cube e a série Prime recebeu seu desfecho apenas no Nintendo Wii. A Retro Studios mais uma vez fez bom uso da capacidade do console, usando de todo o potencial do Wii para nos dar um jogo brilhante. Visualmente é tão lindo quanto Prime e Echoes, mas sua verdadeira exclusividade está no uso do Wiimote+Nunchuk. Corruption usa o pointer do controle para mirar, o que também dispensava o uso de mais de um analógico. O sensor de movimento também era usado para pular com a Morph Ball e para usar o Grape Lasso. A precisão dos controles do Wii tornaram o game um dos obrigatórios do console. Corruption continuou com a aparição de novos Hunters e a livre escolha entre os planetas que o jogador vai visitar, qualidades importadas do Metroid Prime Hunters. O enredo também é mais complexo e fecha de maneira exemplar a trilogia Prime. Em Corruption, Samus e mais três Hunters recebem a missão de recuperar as Aurora Unit, mas o jogo dá um giro quando um ataque dos Space Pirates quase corrompe todo o planeta Norion com Phazon. A aparição da Dark Samus e o fato dos Hunters terem sido infectados com Phazon faz com que os quatro tenham de usar armaduras especiais, capazes de usar a substância para aumentar o poder do usuário. Samus só volta à consciência semanas depois, muito depois de seus companheiros. Ela então é mandada para investigar o desaparecimento dos outros caçadores e destruir os Leviathans, cometas recheados de Phazon, que caíram em planetas próximos. Metroid Prime: Corruption foi o último jogo de Metroid a ser lançado até então. Do dia 27 de Agosto de 2007 para cá, ou 6 de Março de 2008 para os japoneses, não recebemos nenhum outro jogo de Metroid fora a compilação dos três jogos em Metroid Prime Trilogy.

CRONOLOGIA

MANGÁS OFICIAIS

METROID

METROID PRIME

METROID PRIME HUNTERS

METROID PRIME 2: ECHOES

METROID PRIME 3: CORRUPTION

METROID II: RETURN OF SAMUS

SUPER METROID

METROID: OTHER M (AINDA NÃO LANÇADO)

METROID FUSION

Venha fazer parte da
Equipe Nintendo Blast!

VAGAS: ***Newsposter, Redator, Designer, Diagramador, Revisor, Videoposter.***

Mais informações

Inscrever-se

ARC RISE FANTASIA

por **Leonardo Marinho**

Arc Rise Fantasia é o próximo RPG a aportar exclusivamente no Nintendo Wii. O game foi produzido pela Image Epoch, a mesma de Luminous Arc, para o Nintendo DS, e conta com trilha sonora do honorável Yasunori Mitsuda, o responsável pelas incríveis músicas de Chrono Trigger e Xenogears. O jogo é um clássico RPG japonês com batalhas baseadas em turnos e gráficos no estilo anime.

O game se passa no Império de Meridian, um reino que sofre com a ameaça dos Feldragons, dragões contaminados e altamente perigosos. Para complicar ainda mais a situação, o corpo destas criaturas é tão tóxico que caso uma dessas bestas seja derrotada a explosão resultante contaminará imediatamente o local de sua morte, transformando-o em um território perigoso e impróprio para a vida.

A história começa quando a capital Diamant sofre um ataque de uma horda de Feldragons e o Império envia seu exército como reforço. Entre os soldados encontra-se L'Arc, um mercenário experiente que conta com a ajuda de Alf, príncipe de Meridian e o segundo na linha de sucessão ao trono. Durante a batalha, L'Arc é ferido e cai da airship que o transportava indo parar no meio de uma floresta. Lá ele encontra uma garota desconhecida que se apresenta como Ryfia, e descobre que ela pertence a um reino atualmente em guera contra o Império. Tendo seus ferimentos curados por Ryfia, L'Arc decide escolta-la até seu lar, sem saber que a garota esconde grandes segredos e dando início a toda a aventura.

Você andará pelo mapa seguindo de cidade em cidade, passando por cavernas que se revelam verdadeiros labirintos, derrotando monstros e chefes, evoluindo lenta e gradativamente seus personagens e gastando seu suado dinheirinho para comprar armas e equipamentos melhores para sua trupe de aventureiros. Um prato cheio para os amantes do gênero.

Os monstros são visíveis nos mapas, o que significa que você não sofrerá a tensão de batalhas aleatórias e ainda pode projetar suas fugas, tornando a exploração bem mais rápida. Nas batalhas suas ações são limitadas pela sua barra de AP. Cada ação que seus personagens realizam consome AP e o grupo inteiro compartilha a mesma barra, que se regenera no começo de cada turno. Você pode determinar exatamente quem fará o quê e o quantas ações realizará, garantindo maior controle dos AP.

As opções são triviais: pode-se atacar, defender, usar uma de suas magias, mover-se ou não fazer absolutamente nada e simplesmente esperar o próximo turno enquanto seus companheiros suam a camisa. Combos podem ser criados usando uma sequência de ataques e o jogador ainda tem a opção de realizar um movimento chamado “Trinity Act”, uma espécie de ataque especial executado em conjunto pelos personagens e que resulta em grandes quantidades de danos.

O game ainda conta com um sistema de Orbs especiais, que são a fonte de magia dos personagens. Cada arma pode ser equipada com essas Orbs, garantindo uma grande variedade na combinação de ataques e poderes especiais. Utilizando-os com sabedoria, é possível tirar vantagem das fraquezas dos inimigos e causar-lhes mais danos.

No que tange a parte gráfica, Arc Rise Fantasia não faz feio. Os cenários são coloridos e as cores bem vibrantes. Entretanto, a maioria dos diálogos se passa em 2D, com o semblante de seus personagens e a caixa de texto na parte inferior. Nada além do que já estamos acostumados. Tudo muito belo e que faz bom uso da capacidade do Nintendo Wii.

A história principal rende, no mínimo, umas 60 horas de jogo e até mais se você se empenhar em terminar todas as side quests. Há ainda uma preocupação significativa com os diálogos secundários, que ilustram com mais detalhes os relacionamentos entre os personagens e explicam melhor certos aspectos da trama. A dublagem americana deixa um pouco a desejar quando comparada a japonesa. E para os mais exigentes, infelizmente a dublagem original não estará disponível nas opções.

Arc Rise Fantasia chega ao Nintendo Wii dia 20 de julho, pouco mais de um ano após seu lançamento no Japão. O game recebeu nota 32 de 40 da publicação japonesa Famitsu e promete não desapontar. Se estiver procurando um bom RPG que lhe renda uma jornada de mais de 60 horas, então fique de olho neste lançamento.

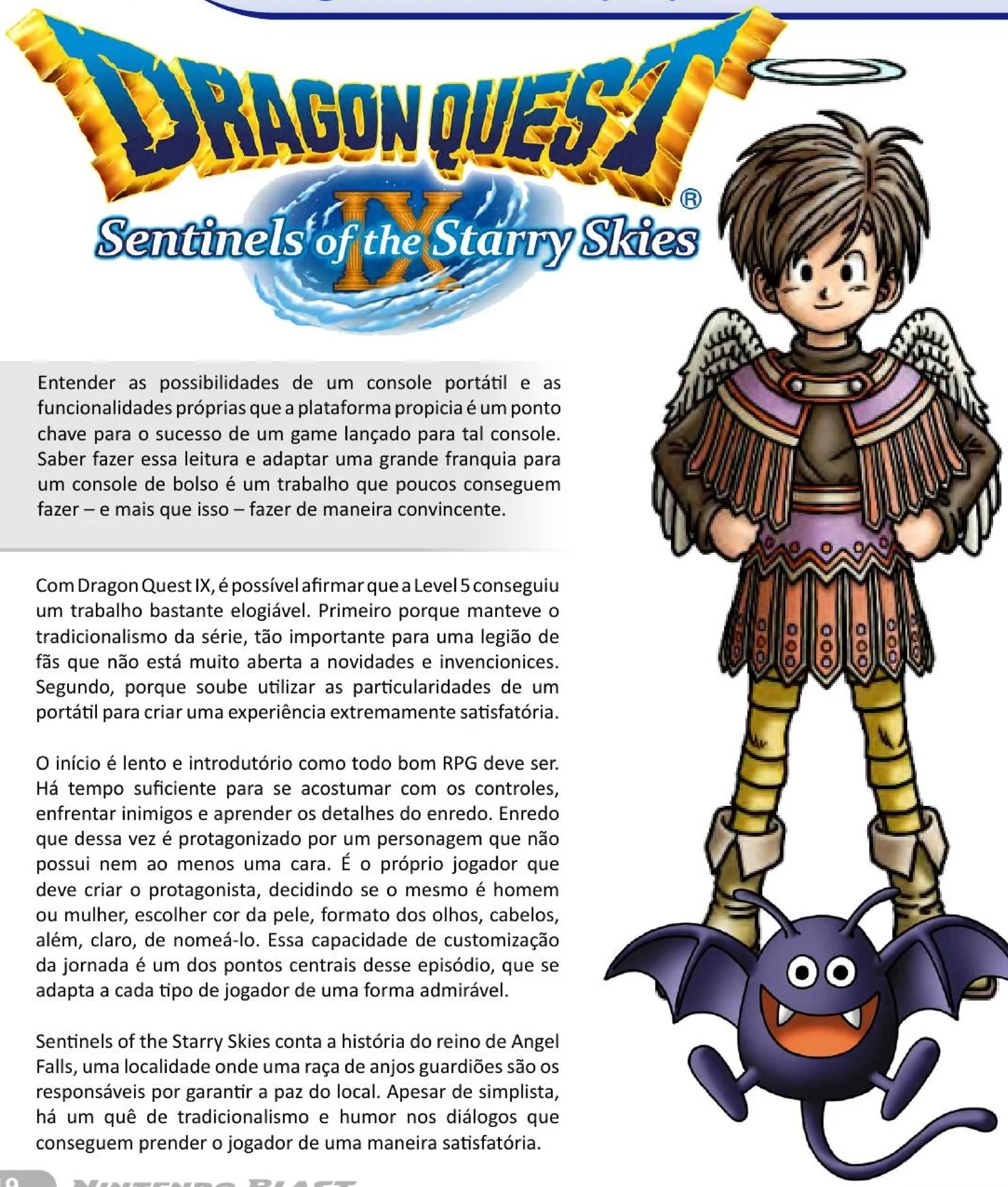
ARC RISE FANTASIA

Gênero: RPG
Console: Exclusivamente no Nintendo Wii
Publisher: Ignition Entertainment
Developer: Image Epoch
Data de lançamento: 20 de Julho de 2010 (EUA)

por Gustavo Assumpção

Análise

Dragon Quest IX (DS)



Entender as possibilidades de um console portátil e as funcionalidades próprias que a plataforma propicia é um ponto chave para o sucesso de um game lançado para tal console. Saber fazer essa leitura e adaptar uma grande franquia para um console de bolso é um trabalho que poucos conseguem fazer – e mais que isso – fazer de maneira convincente.

Com Dragon Quest IX, é possível afirmar que a Level 5 conseguiu um trabalho bastante elogiável. Primeiro porque manteve o tradicionalismo da série, tão importante para uma legião de fãs que não está muito aberta a novidades e invencionices. Segundo, porque soube utilizar as particularidades de um portátil para criar uma experiência extremamente satisfatória.

O início é lento e introdutório como todo bom RPG deve ser. Há tempo suficiente para se acostumar com os controles, enfrentar inimigos e aprender os detalhes do enredo. Enredo que dessa vez é protagonizado por um personagem que não possui nem ao menos uma cara. É o próprio jogador que deve criar o protagonista, decidindo se o mesmo é homem ou mulher, escolher cor da pele, formato dos olhos, cabelos, além, claro, de nomeá-lo. Essa capacidade de customização da jornada é um dos pontos centrais desse episódio, que se adapta a cada tipo de jogador de uma forma admirável.

Sentinels of the Starry Skies conta a história do reino de Angel Falls, uma localidade onde uma raça de anjos guardiões são os responsáveis por garantir a paz do local. Apesar de simplista, há um quê de tradicionalismo e humor nos diálogos que conseguem prender o jogador de uma maneira satisfatória.

Próxima

A jornada fica bem longe do lugar comum. Há várias reviravoltas ao longo da história, justificando as muitas buscas por itens e a jornada longa e funcional. Esse tradicionalismo ainda é mais visível na maneira com que o game se desenrola. O sistema de progressão da história é típico dos JRPGs, exigindo uma grande capacidade de paciência e dedicação para enfrentar a linearidade que a história acaba propondo.

Colete, batalhe, ganhe novas habilidades

Quem está acostumado com Dragon Quest deve se lembrar de como é chato ter que conviver com batalhas aleatórias. Além de maçantes, elas ainda podem ser fatais em um momento no qual o jogador está desprevenido – sem itens de cura por exemplo. Pensando nisso, a Level 5 excluiu esse sistema dessa nona versão. Agora os inimigos são visíveis na área de jogo, embora continuem aparecendo em grande quantidade – ideal para aquele momento onde é preciso subir alguns níveis.

O sistema de batalhas não poderia ser mais simples. Ao se encontrar um inimigo, o jogador é transportado para um menu, onde deve escolher as ações que cada um irá desempenhar no turno. É possível escolher ataque, defesa, magia ou então utilizar um item. Após escolhidas as ações, a tela superior mostra o resultado delas. O esquema, como deu para perceber, é bem parecido – ou até mesmo idêntico – aos demais games da franquia, o que pode ser encarado como uma decisão segura da equipe de desenvolvimento. Mesmo assim, há um certo desconforto, já que melhorias para tornarem esses confrontos menos mecânicos poderiam ter acontecido. É difícil esconder o quão ultrapassado esse esquema é.

Há algumas coisas elogiáveis no sistema dessa nona versão. A primeira é a existência de classes para os personagens – que aqui são chamadas de vocations. São elas que adicionam uma grande quantidade de estratégia, já que cada personagem pode desempenhar um papel diferente durante as batalhas: Warrior, Mage, Priest, Thief, Martial Artist ou Minstrel. Aliás, Minstrel é a classe base, e a partir dela é possível alterar para outras. Classes destraváveis também estão disponíveis. Ter um grupo com a presença de classes variadas é um dos pontos chave para se obter sucesso na jornada.

Multiplayer viciante

Onde Dragon Quest IX realmente brilha é em suas capacidades multiplayer. O game permite até quatro jogadores simultâneos, sendo que um deles atuará como host e outros três como convidados. É possível enfrentar batalhas em grupo, ou então enfrentar batalhas separadamente – cabendo aos próprios jogadores decidirem tal feito. Há a presença de quests especiais para o jogo multiplayer, já que não é possível avançar na história principal nesse modo. Apesar disso, a disputa com amigos compensa, já que se torna um refresco em meio a progressão exigente da história principal.

Outro ponto interessante é a existência de múltiplas side-quests e eventos paralelos, responsáveis por uma ampliação considerável da experiência de jogo. Já estão disponíveis para download as primeiras quests extras e outras dezenas devem ser disponibilizadas ao longo do tempo.

O frescor do título fica pela existência de um sistema de criação de itens que é impressionante, além de muito útil. É possível coletar matéria-prima ao longo do jogo para fabricar suas próprias armaduras, lanças, espadas e tudo mais que for permitido. Receitas são adquiridas em livros encontrados em bibliotecas e masmorras, de certa forma obrigando o jogador a vasculhar cada canto dos cenários.

O medievo em duas telas

Como sempre, a ambientação de Dragon Quest IX é tipicamente medieval, havendo a existência de arquétipos típicos dessa representação. O desenho de inimigos e personagens é bastante convincente, já que é resultado do trabalho sempre acima da média do mestre Akira Toriyama (assim como o que já acontecia em outros títulos da série).

O visual é muito interessante, embora não seja brilhante. A aplicação do estilo cel shading nos gráficos combina perfeitamente com a proposta de um portátil, oferecendo um visual bastante competente. Há um bônus importante: ao se equipar roupas, itens, armaduras, seu personagem também irá mudar de aparência, um recurso poucas vezes visto no DS. Um pouco preocupante, é que não são raras as vezes em que o jogador pode se sentir jogando uma versão portátil de Dragon Quest VIII (lançado em 2005 para o Playstation 2), tamanha é a semelhança visual entre os títulos (salvo as devidas proporções técnicas, claro).

Essa sensação de Deja Vú também acaba atingindo a parte sonora do título. A trilha é pouco inspirada e remete ao trabalho de games anteriores da franquia. Outro destaque negativo é a inexistência de qualquer atuação de voz, o que, convenhamos, é inaceitável para os padrões atuais. Há de se considerar a preocupação com a manutenção do tradicionalismo típico de Dragon Quest, mas não deixa de ser um contra.

Entender o valor da tradição. Esse é o principal requisito para aproveitar Dragon Quest IX em todas as suas particularidades, apesar do estilão old-fashioned que pode incomodar alguns. A jornada é longa, heróica, homérica. Só quem saber reconhecer esses valores como inerentes ao próprio universo da série vai conseguir aproveitar ao máximo essa aventura. E essa jornada tem feeling, pode apostar.

8.5

DRAGON QUEST IX- NINTENDO DS

GRÁFICOS **8.5** | SOM **7.5**
JOGABILIDADE **8.5** | DIVERSÃO **9.0**

TOP 10 Grandes Promessas

Colaboração:
Eduardo Jardim
Sérgio Estrella

por **Gustavo Assumpção**

NINTENDO 3DS™

O anúncio do Nintendo 3DS na E3 foi uma grande surpresa, principalmente por um aspecto: os games. Fazia muito tempo que um console não conseguia logo de estreia atrair tanto apoio de tantas produtoras diferentes. Passado a E3 2010, elaboramos uma lista com as dez maiores promessas do portátil entre todos os games anunciados. Confira e entre no hype você também!

10 Professor Layton and the Mask of Miracle

A série Professor Layton talvez seja um dos grandes fenômenos dos últimos tempos. Criação da Level 5, já rendeu quatro games diferentes, sendo que apenas os dois primeiros deles (Curious Village e Diabolical Box) chegaram ao ocidente. O terceiro game, Unwound Future, também já está confirmado para lançamento por aqui.

Quando todo mundo imaginava que a série continuaria sua trajetória ainda no DS, eis que a E3 2010 nos revelou uma grata surpresa: o quinto game da franquia será lançado para o Nintendo 3DS. Os detalhes sobre essa continuação ainda são bem escassos, mas já é possível fazer várias especulações sobre o que essa continuação pode trazer - principalmente com relação a sua trama.

Mask of Miracles é o quinto game da série, mas o terceiro se levarmos em conta a ordem cronológica. Ele se passa pouco antes do quarto game, Professor Layton and the Specter's Flute (ainda sem data pra chegar ao ocidente). Layton e Luke chegam à Casino City, cidade criada pelos poderes da misteriosa Mask of Miracle. Lá, terão que explorar a cidade para descobrir a verdade por trás da máscara. A máscara pode conceder desejos, mas também tem sido associada a vários incidentes misteriosos. Já está confirmado o retorno de alguns personagens de outros games da franquia - além claro dos novos, que vão apimentar a história. O jogo deve fazer uso dos novos recursos do portátil, como as imagens 3D e os sensores de movimento, para apresentar novos puzzles ao jogador.

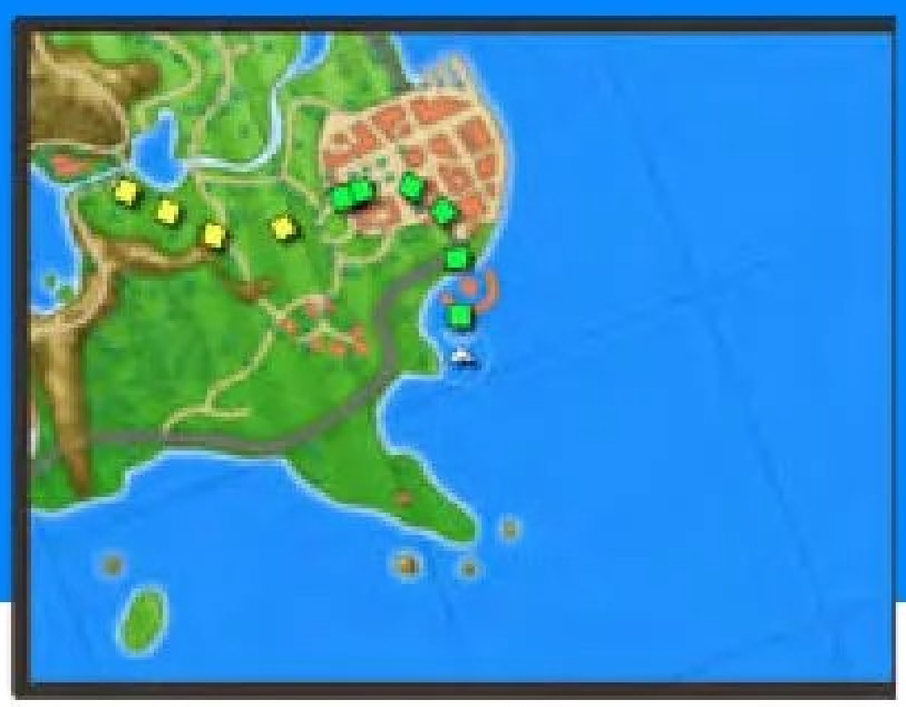
9 Paper Mario 3DS

Uma das maiores novidades durante a E3 2010 para os fãs de Mario foi o anúncio de um título da série Paper Mario para o 3DS. O jogo marcará a primeira vez em que a série Paper Mario decolará num console portátil, ignorando a regra de divisão das linhas de RPG do encanador: Paper Mario para consoles de mesa, e Mario & Luigi para consoles portáteis. Esse costumava ser o padrão, pelo menos até agora

Com o pouco que vimos de Paper Mario 3DS (título ainda não confirmado, pode mudar ao longo do ano) já temos uma ideia básica de como será o jogo: o sistema de batalha clássico que pende para um estilo de plataforma (que aconteceu em seu predecessor para Wii, Super Paper Mario), será deixado de lado para dar lugar ao retorno do sistema de luta baseado em turnos dos dois primeiros jogos da série (Paper Mario para Nintendo 64 e Paper Mario: The Thousand-Year Door para GameCube). Os clássicos e originais comandos de pulo (Jump) e martelo (Hammer) estão presentes, com a adição de um layout reprojetado para a touch screen do 3DS. Algumas screenshots mostram que parceiros retornam após terem sido substituídos por Pixls, tais como um aterrorizante Chain Chomp, que é visto ajudando o encanador a derrotar os oponentes. Outras screenshots mostram objetos caindo nos lugares, nos induzindo a deduzir que os jogadores terão de mover certos objetos ao redor da tela para resolver quebra-cabeças. Também é interessante notar que, para fazer boa demonstração da visualização tridimensional, inimigos poderão ser lançados contra a tela, dando a impressão de que estão sendo nocauteados para fora.

Mario é o único protagonista visto até agora, e os inimigos incluem Goombas, Pokeys, Koopa Troopas Vermelhos, Whomps e o Rei das Toupeiras Monty (o último, carrega a força para arremessar rochas gigantescas contra os heróis). Uma fase que lembra o World 4 de New Super Mario Bros., uma sala repleta de cartas gigantes de baralho, pontes movediças, enfim... muita coisa foi reservada para Paper Mario para 3DS. Vamos esperar!

8 PilotWings Resort



Parece que faz pouco tempo, mas a série PilotWings está há muito tempo desaparecida. O último game da série a chegar as lojas foi o famigerado PilotWings 64, em 1996, no ano do lançamento do Nintendo 64.

Após 14 anos de espera, a série PilotWings finalmente ganha uma nova edição, desta vez no 3DS. Mas quem esperava por algo totalmente inédito pode se decepcionar, já que o novo jogo da série é baseado no modo Airplane de Wii Sports Resort (imagem acima) – que por sua vez baseou-se na mecânica de PilotWings. Isso não significa necessariamente que o game será algo menor, já que a Nintendo está trabalhando duro para que a experiência oferecida no 3DS seja ampliada e refinada - ainda mais porque agora ele não é algo mais de um pacote e sim o próprio game.

O game para 3DS incluirá até mesmo a Wuhu Island, ilha fictícia que também esteve presente em Wii Fit Plus. Ainda não ficou claro quais mudanças a Nintendo trará ao título, além do suporte ao 3D, é claro, e aos Miis. Vale lembrar que o acelerômnetro do 3DS deversá ser o ponto central da jogabilidade.

7 Animal Crossing

Pouco antes da E3, já se esperava o anúncio de um Animal Crossing para o Nintendo 3DS. A informação de que um game da série estaria disponível logo no lançamento do portátil foi uma das principais especulações do período pré-E3.

Pois bem, com a confirmação da versão, muito tem se dito sobre as possibilidades da franquia no portátil da Nintendo. Pelas primeiras imagens divulgadas, é muito complicado fazer qualquer suposição sobre as mudanças que a produtora pretende - mas é lógico que não vamos deixar de especular muito em torno dessa sequência.

Nós já sabemos que o 3DS terá uma série de funcionalidades online ampliadas - como a presença de uma maior quantidade de recursos para o jogador. - ampliando muito o potencial que Wild World (lançado para o DS em 2006) já mostrou. O download de conteúdo extra também deverá se tornar um dos grandes filões do portátil - e por extensão dos games que serão lançados para ele. Outro elemento que parece fazer sentido na mecânica de Animal Crossing seria a possibilidade de fotografar objetos para que eles apareçam de forma tridimensional - graças as poderosas câmeras do portátil.

É possível perceber que a evolução no visual parece ser a única certeza até agora. E esperamos sinceramente uma quantidade suficiente de novidades para evitarmos a frustração que foi o mais do mesmo Animal Crossing City Folk.

6 Nintendogs + cats

nintendogs +cats™

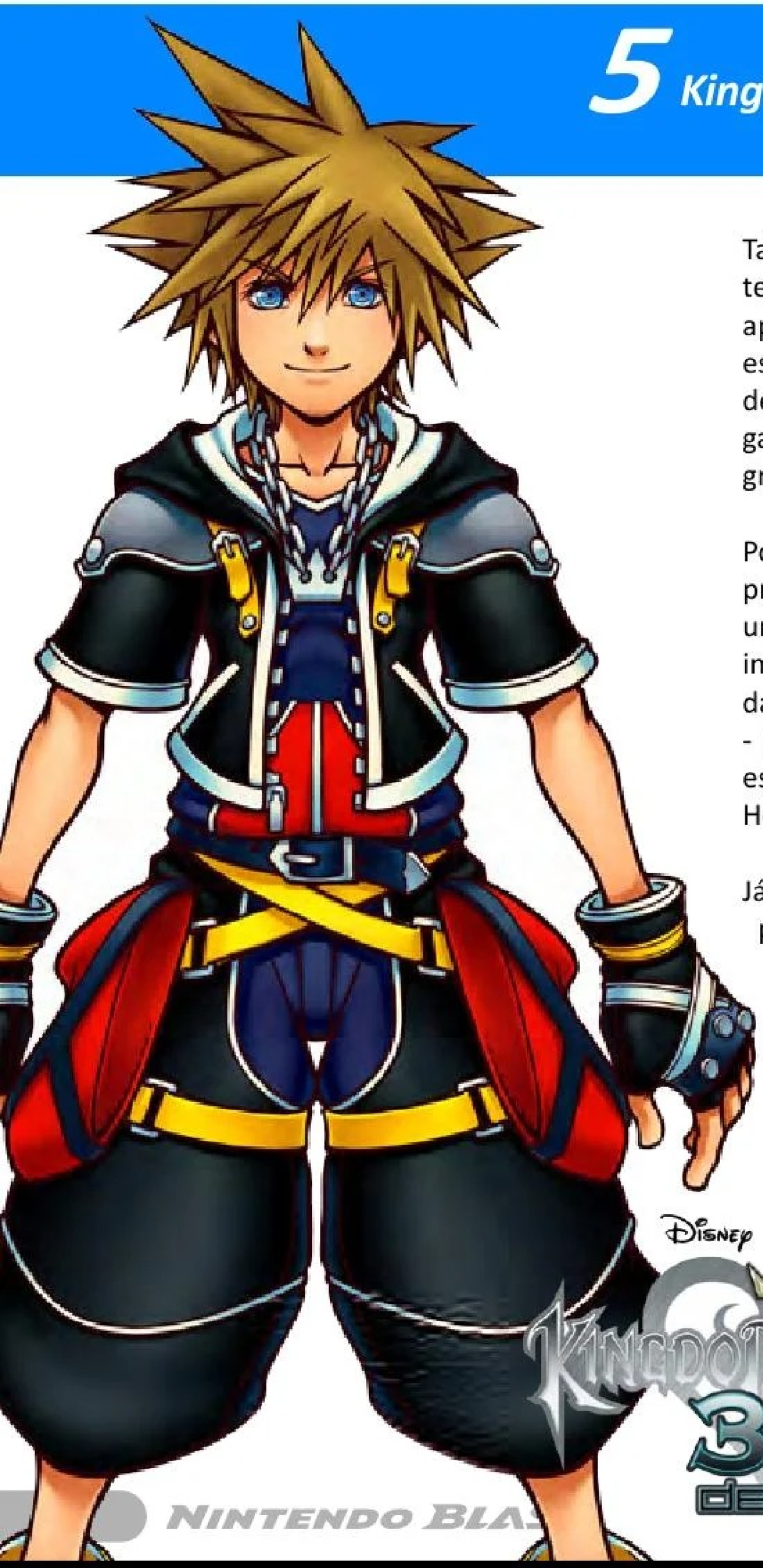
Um dos grandes responsáveis pelo sucesso do Nintendo DS atende pelo nome de Nintendogs. Após ganhar nota máxima na publicação japonesa Famitsu, o simulador criado pela mente de Shigeru Miyamoto se tornou um enorme sucesso de público e crítica. Mas, ao contrário do que se esperava anteriormente, o título nunca recebeu uma continuação. Essa injustiça deixa de existir com o lançamento do Nintendo 3DS, já que Nintendogs + cats é um dos principais títulos da primeira leva do portátil.

E logo as primeiras informações sobre o game o tornam simplesmente imperdível. Segundo a Big N, a principal novidade acaba sendo o uso de uma das principais funcionalidades do portátil: a câmera embutida. Nintendogs + cats utiliza uma tecnologia de reconhecimento facial para permitir que os filhotes reajam aos movimentos reais, imitando a inclinação de sua cabeça e lambendo a tela quando você se inclina para ela. Esse é o mesmo preceito de interação presente no Kinect - ou seja, a Nintendo mais uma vez está atenta ao que seus concorrentes estão fazendo.

Outra inovação considerável vem do visual evoluído, mostrando uma adição de efeitos e recursos. Os cães agora são muito mais animados e vivos e os cenários utilizam os efeitos de profundidade de uma maneira bastante interessante. Vale lembrar que o uso do 3D permite uma maior percepção desses efeitos, potencializando um trabalho muito interessante.

Dá pra esperar um uso interessante das funcionalidades online do 3DS - e por que não, do acelerômetro. Saberemos mais novidades em breve.

5 Kingdom Hearts 3DS



Talvez a demo de Kingdom Hearts presente na E3 2010 tenha sido a grande surpresa entre todos os games apresentados para o portátil. Mesmo que tenha mais escondido do que mostrado, o título conseguiu se destacar em meio a uma verdadeira enxurrada de games anunciados - sendo que alguns deles foram de grande peso.

Pois bem, hoje já conhecemos um pouco sobre o novo projeto, que será um jogo completamente novo - e não um remake do primeiro título, como muitos podem imaginar pelas imagens. Esse vai ser o próximo passo da franquia após re: Coded, 358/2 Days e Birth by Sleep - provavelmente funcionando como uma ponte entre esses spin-offs e um possível (e provável) Kingdom Hearts III.

Já foi confirmado que o game terá Sora e Riku como personagens jogáveis e que ambos irão visitar novos mundos. Aparentemente, haverá uma conexão entre as histórias de 3DS e Re:Coded. Ou seja, apesar de ser um spin-off, o game terá uma importância crucial na trama da série - já que essa versão é protagonizada por Sora e Riku, dois dos mais importantes personagens de toda a trama criada pela Square Enix.

É lógico que esse é um daqueles games que pouco vão se aproveitar do 3D, já que o visual não é o objetivo principal da série. Mas é certo que dará um gás novo.

NINTENDO BLA

4 Mario Kart 3DS

Não tem jeito, virou regra. A cada novo portátil que chega as lojas, a Nintendo lança um novo Mario Kart. Foi assim no GBA, no DS e agora no 3DS. E isso não é algo que devemos criticar, muito pelo contrário. Só que não dá pra deixar de afirmar que a principal inovação em cada game da franquia é o update visual.

Isso não é muito diferente aqui. O que mais chamou a atenção nesse novo game da franquia é o cuidado visual com os detalhes. Feito para um hardware mais poderoso, o título oferece um visual cheio de efeitos e animações novas. É um Mario Kart DS turbinado.

Se do visual sabemos algo pelo que vimos, o mesmo não podemos dizer dos recursos e funcionalidades dessa versão. Até o momento, a Nintendo fez segredo e não revelou quase nenhuma informação sobre o que o game trará de novo. É óbvio que dá pra especular, já que sabemos boa parte do que portátil vai oferecer. Espere conquistas, um modo online super completo e coisas do tipo. E prepare-se para perder boa parte da sua vida nos meses seguintes ao lançamento...

3 Metal Gear Solid Snake Eater 3D "The Naked Sample"

A revelação de que um Metal Gear Solid estava sendo desenvolvido para o Nintendo 3DS pegou muita gente de surpresa. MGS3D era o projeto secreto em que Hideo Kojima estava atuando como diretor, conforme o próprio alardeava via Twitter – e aliviado, confirmou após o término da conferência da Nintendo.

Mais tarde, o designer revelou que o desenvolvimento da demo foi secreta até mesmo dentro da própria Kojima Productions "Foi um projeto secreto, por isso mesmo até na Kojima Productions ninguém sabia sobre ele, embora nós tenhamos feito isso com a equipe de Peace Walker".

Ele ainda revelou que desenvolveu uma demo maior do que a demonstrada na E3, mas a Nintendo pediu que a mesma fosse encurtada para que mais pessoas pudessem testar. Sobre o visual, um comentário interessante: "Os modelos são de alto nível na mesma qualidade que as coisas que fizemos no PlayStation 3, embora seja difícil ver isso. Se tivéssemos um pouco mais de tempo, poderíamos ter feito algo melhor."

Mas talvez o mais importante foi a confirmação de que apesar da demo apresentada ter sido basicamente técnica, o game está sim em desenvolvimento para o portátil: "O que nós queríamos fazer neste momento é mostrar que nós estamos fazendo uma entrada na franquia Metal Gear Solid que pode ser apreciado em 3D sem óculos de proteção no 3DS.".

Para encerrar, Kojima deu algumas ideias do que pode aparecer no projeto final. Segundo ele, elementos de Peace Walker como o jogo cooperativo deverão estar presentes, apesar de não deixar de ser uma versão melhorada do terceiro MGS "A história de MGS3 não pode mudar, por isso estamos pensando em Co-ops, no 3D e elementos no estilo do portátil".

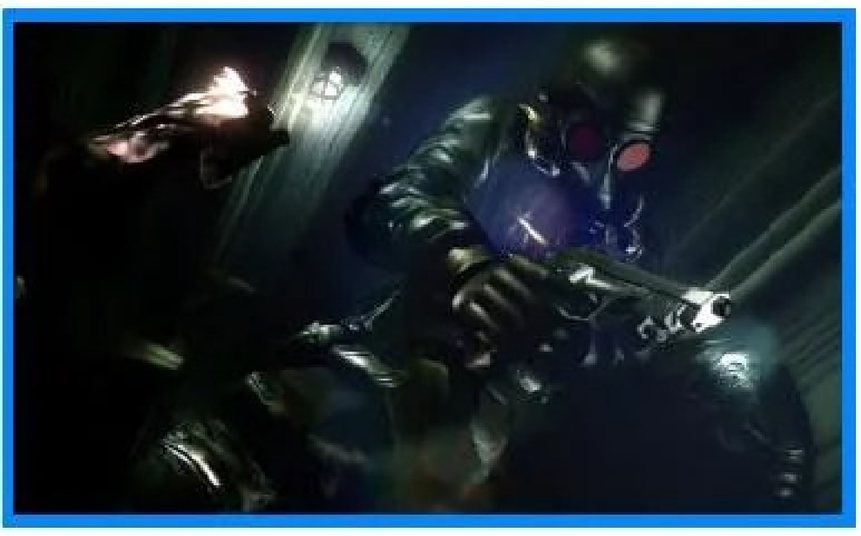
Não tem como não dizer: além de supreendente, esse game para o 3DS quebra um paradigma importante: a inércia da Kojima Productions quando se trata de plataformas da Nintendo.

2 Resident Evil Revelations

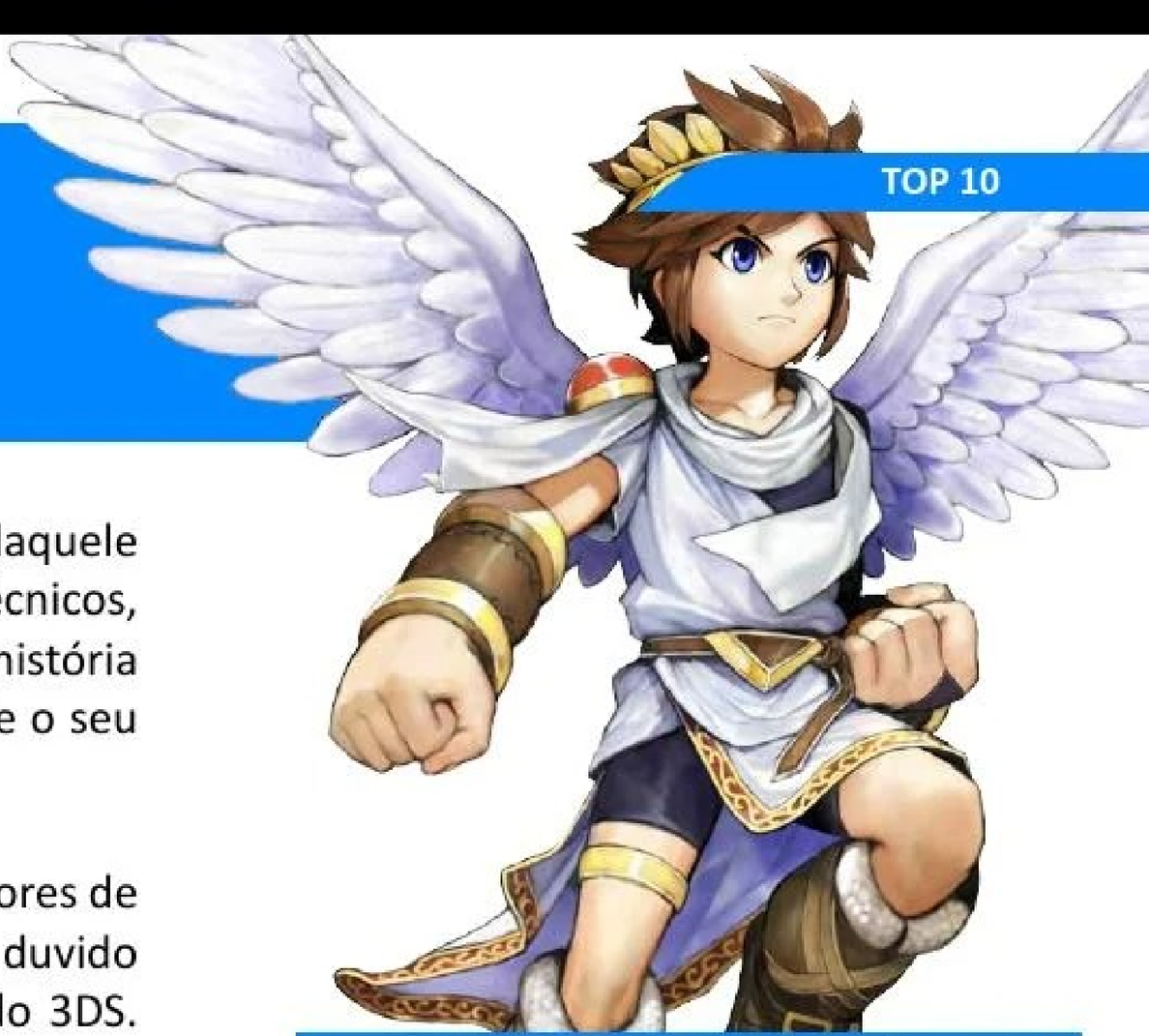
Talvez o game mais bonito apresentado nessa primeira leva do portátil, Resident Evil Revelations é uma das grandes promessas entre todos os games apresentados para o 3DS até agora.

Utilizando a engine MT Framework, a mesma de Resident Evil 5 e Lost Planet 2, o que mais chama a atenção é o visual riquíssimo. Após a E3, chegou-se a cogitar que a demo apresentada não fosse em tempo real, tamanho a riqueza do que apareceu. A Capcom tratou de negar os boatos e confirmou que o game era gerado exclusivamente pelo hadware do 3DS. Um fato não menos que impressionante.

Pelo que foi apresentado, a ação acontece em um navio de carga gigantesco durante uma tempestade forte em alto mar. Chris Redfield e Jill Valentine estão no navio e enfrentam zumbis bem tradicionais. Há grandes chances desse episódio ser o responsável pela ligação entre o quinto e o sexto games da série principal. Detalhes sobre a jogabilidade ainda são completamente desconhecidos, mas não espere muita mudança. Sem dúvida, promete muito!



1 Kid Icarus: Uprising



Em 1986, o NES recebia um dos games mais festejados daquele ano: Kid Icarus. Apesar de enfrentar alguns problemas técnicos, o título é até um hoje um dos mais lembrados de toda a história do console. E olha que já se passam mais de 20 anos desde o seu lançamento.

Nos últimos três anos, são incontáveis as vezes em que rumores de uma possível volta da série pipocaram na imprensa. Mas duvido que alguém imaginava que o retorno se daria no Nintendo 3DS. E mais: que esse é o tão falado game integrante do Project Sora, criado ano passado pelo presidente da Nintendo, Satoru Iwata, juntamente com Masahiro Sakurai.

Apresentado durante a E3 2010, Kid Icarus agradou. Utilizando o mesmo visual do personagem na série Super Smash Bros., o game se tornou um belíssimo shooter, com foco no voo e nas batalhas aéreas. Existem duas possibilidades de ataque: físicos e por tiros. Quando os inimigos estão próximos, um simples aperto de botão irá fazer o ataque físico enquanto múltiplos apertos farão um combo. O mesmo serve para o tiro.

Sobre as armas, haverá basicamente munição infinita. Ao pressionar o botão uma vez, um grande tiro é lançado, enquanto muitos tiros menores saem ao segurar o botão por alguns segundos. Não sei se você notou durante o trailer, mas haviam algumas flechadas que se direcionavam ao inimigo, mudando sua rota. Existe esse tipo de flexibilidade quanta as armas. Diferentemente de um first-person shooter, onde você precisa mirar para alvos pequenos e específicos, tem uma espécie de mira automática, mas uma espécie de elemento corretivo.

Com tantas novidades, não é de se surpreender que esse é o game mais interessante do 3DS até o momento, principalmente porque temos a certeza que está em boas mãos.

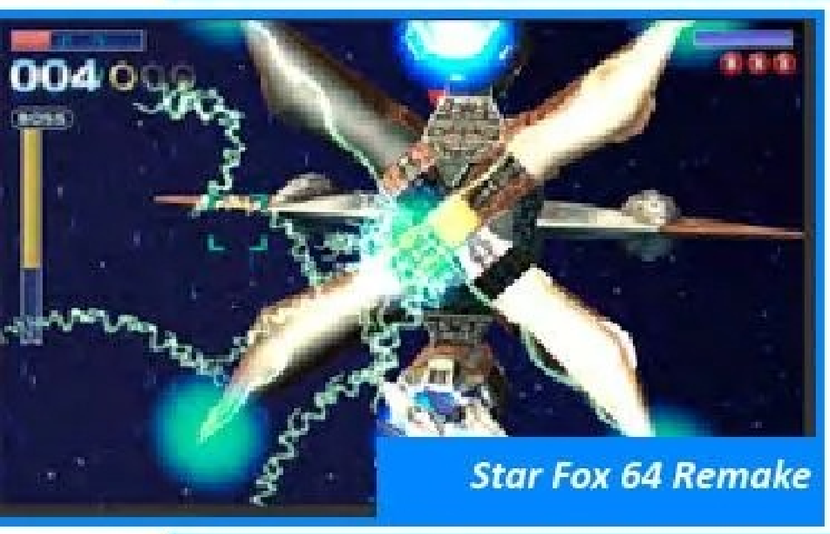
E tem muito mais!

O número de títulos divulgados para o 3DS é tão grande, que muitas promessas ficaram de fora. Fique de olhos bem abertos com eles!

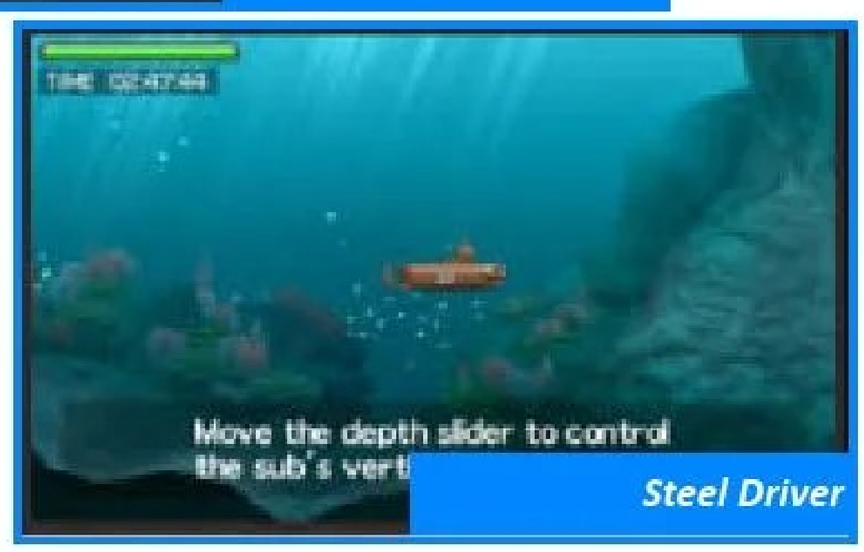
Street Fighter IV 3DS

Zelda: Ocarina of Time

Chocobo Racing 3DS



Star Fox 64 Remake



Steel Driver

Qual game você mais espera jogar no 3DS?

Enquete realizada no Twitter oficial do Nintendo Blast: ***@nintendoblast***

Samusam @nintendoblast Kid Icarus, pq só teve pra NES e claro Zelda OoT, nada melhor q um clássico da Nintendo com gráficos superiores da era 3D.

about 1 hour ago via web in reply to nintendoblast

Manoantunes @nintendoblast tem o Goldeneye 007 também...gsoto de todos esses classicos do N64

about 1 hour ago via web in reply to nintendoblast

tarcisiotm @nintendoblast Dos confirmados, o Ocarina of Time 3D, dos boatos, o possível Super Mario World em 3D. Ambos pela nostalgia =D

about 1 hour ago via web in reply to nintendoblast

Manoantunes @nintendoblast Star Fox 64....sempre adorei esse jogo...tenho ainda hoje...

about 1 hour ago via web in reply to nintendoblast

hudtechhud @nintendoblast RESIDENT EVIL REVELATIONS, resident é resident né e sempre bem vindo em qualquer plataforma... vlw

about 1 hour ago via web in reply to nintendoblast

leandrolopesp @nintendoblast wario. Sempre exploram as possibilidades e ele praticamente define como serão todos os jogos

about 1 hour ago via Twitter for iPhone in reply to nintendoblast

Carurosu @nintendoblast Kid Icarus, o gameplay dele parece instigante e os gráficos 3D vão beneficiar o "mundo aberto" do jogo!

about 1 hour ago via web in reply to nintendoblast

kellvilhena @nintendoblast Resident Evil 3DS *-* Pq ele está muito lindo. só O resto não me deixa muito ansioso.

about 1 hour ago via web in reply to nintendoblast

hugorcampello @nintendoblast Pokemon B&W. pq com a capacidade D, qro me sentir realmente dentro do jogo e poder ter uma maior interação com o game!

about 1 hour ago via web in reply to nintendoblast

Teo1UP @nintendoblast mario galaxy 3ds!

about 1 hour ago via Power Twitter in reply to nintendoblast

Leonixy @nintendoblast Zelda Ocarina of Time 3D, pois eu quero reviver os belos monentos do 64 em 3D e na palma da minha mão xD

1 minute ago via web in reply to nintendoblast

DIVULGAÇÃO

www.nintendoblast.com.br

Já conhece o Nintendo Blast?

O Nintendo Blast é um site de games independente cujo principal foco é a Nintendo, seus consoles, portáteis e franquias. Publicamos diariamente conteúdo de qualidade, que incluem notícias, análises, prévias, dicas e reportagens especiais. Acesse e divirta-se!

METROID™ Other M

por ***Rafael Neves***

Durante uma E3 repleta de continuações incríveis (como Red Steel 2 e Super Mario Galaxy 2), jogos casuais (Wii Sports Resort e Wii Fit Plus, por exemplo), ressurreição de séries por anos esquecidas (podemos citar Golden Sun DS) e até projetos repletos de mistérios (como o novo Zelda Skyward Sword, que na época nem tinha o título definido) eis que a conferência da Nintendo apaga as luzes e tudo o que se vê é um telão mostrando o espaço sideral. De repente, aquele céu escuro e estrelado parece estar sendo bombardeado, como se uma batalha galáctica ali se desenrolasse. E, de repente, o logo da Nintendo surge de uma das explosões. Os espectadores já estavam entendendo mais ou menos o que era aquilo, mas ainda não conseguiam compreender o porquê do logo Team Ninja estar ao lado do da Big N. Então, antes que qualquer um pudesse supor algo, uma base espacial aparece, onde uma bela mulher loira é vista. Rapidamente, ela veste uma armadura conhecida por muitos, não restando mais dúvidas de que era Samus Aran. Então, cenas subsequentes de um jogo de ação frenética enchem os olhos de todos, com Samus desferindo golpes nunca antes vistos em inimigos e cenários variadíssimios. Um verdadeiro show visual, diga-se de passagem. Não era nada e menos nada mais do que **Metroid: Other M**.

Metroid consagrou-se como verdadeiro pilar da Nintendo, batendo de frente com Zelda, Mario e Pokémon. A trajetória da série é repleta de lombadas: Em alguns momentos a franquia foi muito popular e ativa, mas houve passagens de total esquecimento. Quem não se lembra da ausência de Metroid no Nintendo 64? A série só não desapareceu pela ponta de Samus em Super Smash Bros. De uns tempos para cá, a série ganhou uma cara nova e conquistou novos públicos, mas sem perder sua essência como uma aventura solitária de uma caçadora de recompensas em planetas obscuros e repletos de monstros num ambiente futurista. A saga Metroid caminha agora com um novo jogo, cuja proposta é muito diferente da de vários outros jogos da franquia. A Nintendo e a Team Ninja irão juntas trazer um novo Metroid, equilibrando diversos conceitos que várias vezes foram antagônicos. Não há como dizer que Metroid: Other M não seja controverso. Nesse especial, vamos analisar cada uma das seis principais polêmicas do jogo.

Metroid Gaiden? Ninja Samus?

A Team Ninja, desenvolvedora a cargo de Metroid: Other M, tem como principal desenvolvimento no mundo gamer a série Ninja Gaiden. Tal franquia é marcada pelo ritmo de ação frenético e dinâmico, onde a câmera faz os mais loucos movimentos para acompanhar o jogador. Esses elementos estão aliados a grandes hordas de inimigos monstruosos e arte digna, tornando os jogos da série Ninja Gaiden espetaculares. Mas espere ai! Esses mesmo elementos supracitados também são vistos nos trailers de Other M! Seria então o novo Metroid uma mera cópia de Ninja Gaiden? Bem, quando Samus apareceu fuzilando inúmeros alienígenas num piscar de olhos e fazendo uso de movimentos nunca antes vistos para por Space Pirates na lona, a coisa parecia muito com um Ninja Gaiden. No entanto, ao invés de um musculoso Ninja carregando uma espada, vimos uma forte moça trajada numa armadura com um canhão embutido no braço.

Essa polêmica já foi muito discutida, mas ambas as empresas, Nintendo e Team Ninja, afirmaram que Other M não é um mero Ninja Gaiden e sim uma experiência nova. Bem, não se pode dizer que é "totalmente nova", afinal, digamos que Other M bebeu de Ninja Gaiden, como bem diz o designer responsável pelo jogo, Yusuke Hayashi: "Antes de qualquer coisa, não é minha intenção criar uma versão de Ninja Gaiden para Metroid com Other M. Nosso objetivo é definitivamente usar nosso conhecimento para desenvolver algo que seja distintamente Metroid, mas com todas as qualidades que podem torná-lo ainda melhor do que as pessoas esperam que seja. Um trailer sendo um trailer contém todas as sequências que são mais voltadas à ação, talvez um pouco mais violento, mas não será o tempo todo assim no game. Eu terei uma versão que vocês poderão jogar muito em breve". Realmente, não é a cara da Nintendo pegar um jogo de sua linhagem principal e usá-lo como mera "versão" de um outro game.

Other M será recheado de ação, mas, segundo Hayashi, não será sempre assim e nós realmente esperamos que, como todo Metroid, as lutas incessantes sejam interrompidas para momentos de minuciosa

exploração dos cenários. Se esse equilíbrio entre ação, exploração e todo o toque Metroid propiciarão uma experiência única ao jogador, só o lançamento dirá.

Side-Scrolling ou First-Person-Shooter?

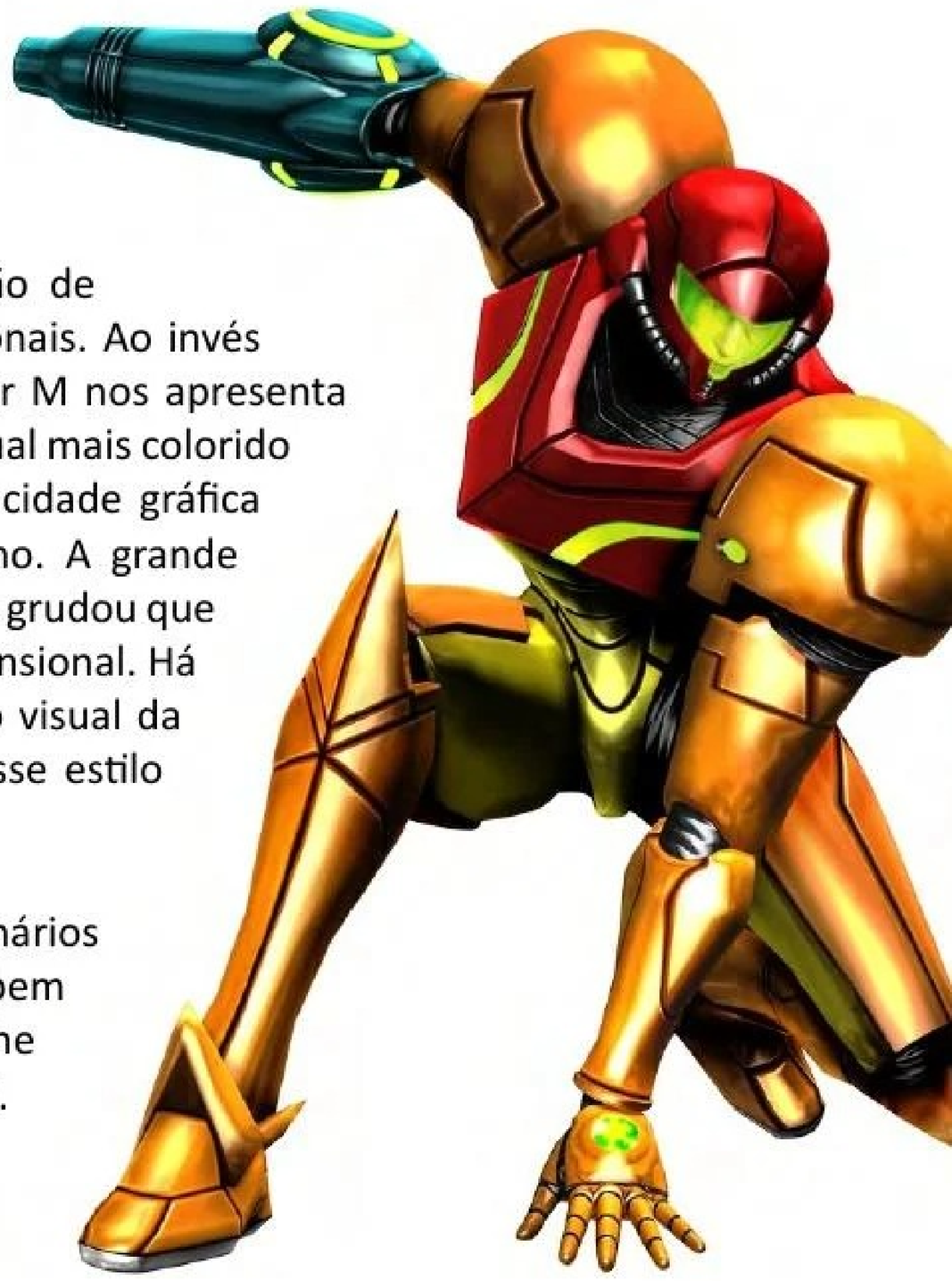
A resposta para a pergunta é: Ambas. Voltemos agora àquele primeiro trailer da E3 de 2009. Primeiramente, Samus era mostrada sob uma perspectiva em 3ª pessoa em cenários bidimensionais e em outros com um grau de profundidade maior, configurando-se como um jogo side-scrolling numa espécie de 2.5D. Mas foi só mudar de cena que Samus foi vista mirando num tentáculo de mais uma criatura alienígena do jogo, e, ao contrário do que era mostrado até então, a visão do jogador estava dentro do capacete de Samus, como em Metroid Prime. Desde ai, uma leva de perguntas surgiu: Quando um jogo é anunciado, uma das coisas que devem ficar claras aos jogadores é a jogabilidade? Foi justamente o contrário que aconteceu em Metroid: Other M. Ao invés de saciada, a comunidade gamer ficou em total alarde, especulando sobre uma possível jogabilidade mista. Bem, pelo menos nisso os palpites acertaram.

A jogabilidade de Other M é de fato diferente, com alternância entre os universos bidimensional e tridimensional, mas bem diferente do que foi visto em Super Paper Mario (Wii). Como já foi mostrado, o jogador controlará Samus com o Wiimote na horizontal, uma posição chave para um jogo em side-scrolling, dando a sensação de realmente estar jogando um Metroid tradicional. Nesse modo, o botão A é usado para se transformar em Morph Ball, e os botões 1 e 2 para pular e atirar, sendo a mira automática. Ações como se pendurar em bordas serão automáticas também, e isso só mostra o quanto os controles de Other M são simples. Sabe aqueles movimentos extraordinários que Samus faz nos trailers? Tipo trepar nas costas de um Space Pirate e atirar bem na coluna vertebral do coitado? Esses golpes são desferidos com o uso do direcional digital, que também é usado para a locomoção da Samus, é claro.

Então, quando for preciso explorar os cenários minuciosamente ou mirar com destreza nos inimigos, o jogador simplesmente virará o controle de forma que

ele aponte para a tela. Quando isso acontecer, a visão do jogador muda e entra dentro do capacete de Samus, assim como fazíamos na trilogia Prime, usando o pointer do Wiimote como mira. Se essa mecânica for precisa e funcional como na série Prime, não teremos dúvidas de que será uma ferramenta poderosa em Other M. No entanto, no modo FPS, Samus não poderá andar, mas apenas mexer a cabeça para os lados. O jogador poderá usar mísseis, o que requer não só apertar um botão, mas levantar o controle como se realmente estivesse carregando um projétil. Infelizmente, alguns jornalistas sortudos que tiveram acesso ao demo do jogo apontaram a transição entre a jogabilidade tradicional e a em primeira pessoa como algo lento e nem tão confortável. Bem, convenhamos que nem toda inovação é perfeita logo de primeira, mas isso não nos deixa menos ansiosos quanto ao game que será Other M. Afinal, ainda há a possibilidade de uma Zero Suit Samus jogável figurar. Como bem disse Yoshio Sakamoto, considerado um dos pais de Metroid: "É difícil dizer agora, mas se for algo que os jogadores realmente queiram, então talvez haja uma possibilidade".

Artisticamente belo ou um retrocesso?



Nem sempre o uso da comparação é o melhor para se analisar um jogo, inclusive porque o que foi mostrado é apenas uma prévia. Às vezes precisamos verificar o histórico de uma série para entender seu atual estágio dela. Metroid é um bom exemplo disso. O visual de Other M remonta aos jogos tradicionais, e isso provavelmente se deve à participação de Yoshio Sakamoto, que trabalhou em muitos Metroid bidimensionais. Ao invés de lindos gráficos realistas, que vimos em Metroid Prime, Other M nos apresenta texturas muito mais vivas e cores mais fortes. Bem, esse estilo visual mais colorido funcionou eficientemente em Metroids 2D, até por que a capacidade gráfica das plataformas passadas não proporcionaria tamanho realismo. A grande polêmica em torno do visual de Other M é se um estilo gráfico que grudou que nem cola nos Metroids bidimensionais funcionará em um tridimensional. Há quem diga que os produtores deveriam ter optado pelo mesmo visual da trilogia Prime, mas também existem aqueles que crêem que esse estilo gráfico é profundo e que funcionará bem.

De certa forma, as animações fluíram bem nos trailers e os cenários obscuros acabaram provocando contraste com efeitos luminosos bem coloridos, característica que funcionou bem em jogos como The World Ends With You (DS). A pergunta é se ficará bom em Metroid. Outro ponto que chama atenção no visual de Other M é a cena

de abertura, que remonta a batalha final de Super Metroid, mas com uma remasterização gráfica que faz inveja até nos consoles HD concorrentes. Ver a Mother Brain, o Metroid que salva Samus sendo reduzido a pó pelo grandalhão, mais monstruosa do que nunca e, no fim, o Hyper Beam se formando no canhão de Samus como um sol é impagável. Na verdade, todas as CGs de Other M parecem ser incríveis e os jogadores que testaram a demo afirmam que a mudança entre jogo e a cutscene é bem sutil. Já foi até afirmado que Other M funciona como uma mistura de filme e jogo, um verdadeiro game cinematográfico. Cenários como bases espaciais, cavernas vulcânicas, florestas tropicais e geleiras foram retratados com uma profunda riqueza de detalhes.

Tudo bem, Other M é belo, mas muitas críticas já foram feitas e a polêmica em torno do visual não para por aí. A armadura de Samus parece ter uma textura muito mais plástica do que metálica, e alguns fãs afirmaram que os inimigos parecem muito mais infantis do que de fato monstruosos. É certo que Other M simplificou os controles, mas isso não quer dizer que o jogo é casual a ponto de tornar os alienígenas fofinhos. Quanto a isso, não há mais o que fazer senão esperar pela vinda do jogo.

Ciborque, Robô ou uma dama?

Parece que aquele susto que levamos ao descobrir que Samus era mulher no final das contas foi simplificado pelo tempo. Quer dizer, foi épico ver uma linda moça saindo de uma armadura ao fim do primeiro Metroid ou ver Samus de biquíni após morrer em Super Metroid, mas será que de fato Samus é uma humana? De que adianta ela fazer tanta questão de ser uma mulher se obedece às ordens da Galact Federation feito um robô e age sem mostrar sentimentos ou qualquer emoção? Bem, para muitos, Samus é uma mera caçadora de recompensas que age conforme ordens de seus superiores. Mas há muitos outros detalhes envolvendo um passado trágico e diversos tormentos ao longo da vida que nem todos os jogadores sabem. E a culpa não é deles, mas sim dos produtores dos jogos, que deixaram esse lado "humano" de Samus literalmente de lado, tendo que recorrer a outras mídias como o mangá oficial para sanar as dúvidas. A proposta de Metroid: Other M é de justamente trazer esse lado emocional da caçadora à tona, deixando-o mais acessível aos jogadores. Como bem disse o diretor Yoshio Sakamoto: "Diferente do resto da série, neste momento

estamos descrevendo o lado humano de Samus através de coisas como animações. Ela é uma mulher forte, mas também tem um lado frágil. Nós queremos fazer um jogo em que haja charme no lado da história e dos toques humanos também. É um jogo de ação, mas é capaz de ter um claro lado emocional".

Tais palavras geraram fortes polêmicas, afinal, Metroid sempre foi uma aventura muito solitária e tenebrosa. Logo, mesclar a solidão à emoção não será uma tarefa fácil. É quase impossível de se prever o resultado, por enquanto. Mas com Sakamoto a cargo do jogo, já podemos esperar por um bom resultado, visto que sua participação em Metroid Fusion trouxe a questão "humana" de Samus à superfície sem perder o louvor do game. Não é a toa que Samus será dublada pela primeira vez na série (tirando os gemidos de dor da saga Prime). Sakamoto fez Samus perder a mudez em Fusion, mas em Other M a caçadora terá longos discursos. Inclusive o jogador poderá ouvir os pensamentos da caçadora, como se eles estivessem na própria cabeça do jogador, imergindo ainda mais nesse lado sentimental de Samus.

Enredo Profundo ou Pretexto para a Ação?

Uma das coisas que ainda permanece envolta em dúvidas é a questão da história de Metroid: Other M. E mesmo com poucas informações, isso já foi o bastante para gerar uma grande polêmica. Ao que se sabe, o novo jogo começa logo no fim de Super Metroid, assim que Samus derrota Mother Brain e escapa do planeta Zebes, antes que ele vire poeira cósmica. Ela é levada ao pronto-socorro com urgência, onde uma equipe da Galact Federation faz seu tratamento. Logo depois disso, uma tarefa é designada a Samus: ela terá de ir até uma base espacial abandonada e explorá-la. Após alguns tutoriais para Samus "reaprender" os movimentos (o que é só uma desculpa para ensiná-los ao jogador), a caçadora chega à estação. Lá, encontra diversos soldados da Federação, incluindo alguns conhecidos como Anthony Higgs (o "Remember me?" do trailer) e Adam Malkevich (lembra daquele computador que te deu dicas e ordens em Metroid Fusion? Ele não é esse computador – óbvio-, mas foi a inspiração para Samus nomear a máquina com esse nome). Ambos estavam ao lado de Samus em missões quando ela ainda era soldado da Federação. Após o combate contra uma monstruosa criatura roxa de tentáculos, Samus aproxima-se mais da equipe, recebendo maiores informações sobre a missão.

Daí em diante, pouco se sabe sobre o enredo de Other M, apenas que ele promete desenvolver melhor os personagens já conhecidos como Samus e Adam e apresentar bem os novos personagens,

como Higgs. Por ter a promessa de ser um jogo profundo, Other M trará boas explicações para vários buracos na cronologia da série, como o intervalo de tempo entre Super Metroid e Metroid Fusion, que não é tão pequeno assim. Por exemplo, será melhor explicado o motivo para Samus ter abandonado o exército, bem como o clima entre ela e seu ex-chefe Adam estar pesado em Other M. Outras polêmicas surgidas no trailers para apimentar a empolgação em torno do jogo serão esclarecidas, como a estranha mulher que se autonomeia Samus Aran no fim do vídeo e quem realmente é Anthony Higgs.

No entanto, há quem queira um Metroid literalmente tradicional em Other M, uma simples desculpa para a ação começar, como "Vá até o planeta 'X' e destrua todos os Space Pirates.". Mas nós sabemos que nem todo enredo básico assim funciona bem na geração atual, principalmente porque Metroid já caminha para uma história complexa desde antes da saga Prime. Já não é de hoje que Metroid tem um enredo bem bolado, e se Other M quer trazer o lado emocional de Samus à tona, provavelmente terá de nos apresentar algo mais complexo e profundo.

Uma Revolução ou um Fracasso Total?

Não há o que se discutir quando falamos que Metroid: Other M apresenta uma proposta de jogo única não só na série Metroid, mas em todo o universo gamer. A última grande polêmica do jogo está no fato desta experiência singular se configurar ou não como uma revolução. Jogos que fogem de modelos tradicionais e que prometem inovar podem ter dois destinos: Sucesso ou fracasso – pois são raros os meio termos. Dentro da franquia Metroid, Other M tem um apelo à complexidade do enredo e ao maior trabalho no lado humano da caçadora de recompensas. Já no amplo mundo gamístico, Metroid: Other M promete trazer uma experiência híbrida entre 2D e 3D incrível e um verdadeiro jogo cinematográfico. Bem, promessas demais poderão levar a um estouro de vendas ou uma decepção completa.

Há quem diga que a retomada de Yoshio Sakamoto à equipe de produção, acoplada às promessas feitas, sejam um sinal de vitória. Mas também existem jogadores esperançosos por uma continuação da série Prime.

E há ainda os mais radicais que jogam a toalha quando o assunto é Metroid: Other M, até porque a Team Ninja nunca fez um Metroid antes, sem merecer tanta esperança na opinião deles. A franquia futurista da Nintendo é muito instável e alguns erros já levaram a grandes depressões, bem como acertos acarretaram em sucesso absoluto. Se a Team Ninja dará mais uma guinada na série Metroid e trará uma nova categoria de jogo ao mundo dos games ou se ela simplesmente arruinará nossas esperanças, não se sabe ainda. É muita polêmica, levando a várias especulações, como a descoberta de segredos e mensagens subliminares no jogo. Já reparou que a sigla do jogo é MOM e que Other M é um anagrama de Mother (ambos siginificam “mãe” em inglês)? Quem sabe não é um indício de que o jogo focará o lado maternal de Samus ou alguma dica sobre a mulher no fim do primeiro trailer, que diz ser a própria Samus Aran. Ou ainda uma possível data de lançamento, já desmentida, escondida no número de mísseis do jogador durante o primeiro trailer? O desenlace está mais próximo do que parece, e não vemos a hora de por as mãos nessa maravilha... ou, quem sabe, nessa decepção.

O sombrio universo de Metroid nos espera mais uma vez! A hora de subir na nave de Samus e partir para planetas totalmente adversos em busca de uma boa dose de exploração recheada com muita ação chegou! Vamos preencher mais uma lacuna no brilhante enredo da saga Metroid e conhecer um pouco mais sobre a caçadora de recompensas mais poderosa da galáxia: Samus Aran.

“Any objections, Lady?”

por **Filipe Gatti**

Developers

Masahiro Sakurai

Quem nunca jogou Super Smash Bros., ou até mesmo algum título da série Kirby, que atire a primeira bola de fogo. São duas franquias monumentais da Nintendo e, na coluna Developers de hoje, falaremos sobre a mente criativa por trás delas.

Masahiro Sakurai nasceu em 1970, em Tóquio, e iniciou cedo na indústria dos games na **HAL Laboratory**, onde aos 19 anos criou o personagem Kirby. Pouco tempo depois houve o lançamento do primeiro jogo com seu personagem, *Kirby's Dream Land*. O desenvolvimento do seu primeiro Super Smash Bros. iniciou anos depois em conjunto com o atual presidente da Nintendo, Satoru Iwata. Lançado em 1999 para o N64, o jogo, que originalmente teve o nome de *Dragon King: The Fighting Game*, se tornou referência na vida de Sakurai.

Depois de muito tempo trabalhando no desenvolvimento de pelo menos nove títulos pela HAL, Sakurai se demitiu em Agosto de 2003, fazendo muitos acreditarem que suas duas séries de grande sucesso, Kirby e Super Smash Bros., não receberiam sequências.

Sua demissão foi consequência de uma insatisfação com a própria estrutura da desenvolvedora e, por incrível que pareça, por estar cansado de criar sequências para as duas franquias que criou. Não muito tempo depois, ainda no mesmo ano, Sakurai se juntou com Tetsuya Mizuguchi, criador de *Rez*, na **Q Entertainment**, onde criou o puzzle *Meteos* para o Nintendo DS.

Apesar da quebra de laços com a HAL, Sakurai trabalhou como diretor e game designer de *Super Smash Bros. Brawl* lançado em 2008, onde também participou fazendo a voz do personagem *King Dedede*.

Em Setembro de 2005 fundou sua própria desenvolvedora, a Sora, onde trabalha até hoje. Recentemente, na E3 2010, a Sora, Sakurai e Nintendo anunciaram o desenvolvimento de Kid Icarus para o Nintendo 3DS.

Curiosidades

Apesar de ser reconhecido pelo lançamento de Kirby's Dream Land, o primeiro jogo em que Sakurai trabalhou foi Arcana, um RPG para Super Nintendo no qual seu nome aparece durante os agradecimentos especiais.

O nome original de Kirby era PoPoPo e a decisão sobre sua cor foi muito debatida. Contrariando a sugestão do próprio Shigeru Miyamoto, que disse que ele deveria ser amarelo, Sakurai decidiu que ele seria rosa. Devido a uma confusão na América do Norte sobre a decisão da cor, ele aparece na capa do jogo simplesmente como branco. O nome de Sakurai também aparece nos créditos de Kirby 64 na parte como um dos dubladores de personagens.

Games em que Sakurai trabalhou

- Arcana
- Kirby's Dream Land
- Kirby's Adventure
- Kirby's Pinball Land
- Kirby's Block Ball
- Kirby Super Star
- Super Smash Bros.
- Super Smash Bros. Melee
- Kirby: Nightmare in Dream Land
- Kirby Air Ride
- Kirby & The Amazing Mirror
- Meteos
- Meteos: Disney Magic
- Kouchuu Ouja: MushiKing – Greatest Champion e no Michi
- Super Smash Bros. Brawl
- Kirby's Epic Yarn
- Kid Icarus Uprising

por Sérgio Estrella

Custom Blast

Um Slogan para o Nintendo Blast!

Nesta edição temos o orgulho de estrear a sessão **Custom Blast**, que promoverá concursos de criação com nossos leitores, seja em imagem, texto, ou outros formatos. Tudo acontece através do Fórum Nintendo Blast e os resultados sairão nas edições da revista. Os vencedores receberão pontos e poderão ganhar prêmios! Acesse http://forum.nintendoblast.com.br e saiba mais - além de poder participar do próximo Custom Blast!

E na primeira edição, o tema escolhido foi: ***Crie um slogan para o Nintendo Blast!***
Confira os resultados e os usuários vencedores:

1º Vencedor **Bruno DS**

NINTENDO BLAST
Se o assunto é Nintendo, nós detonamos.

2º **madmax77**

NINTENDO BLAST
Com você até o game over!

3º **RedNiNero**

NINTENDO BLAST
Quem lê, DETONA!

Menções honrosas:

"Dê 1-up em seu conhecimento sobre Nintendo." ***(MatMario)***
"Pensou Nintendo? Pensou no Blast!" ***(Jarmin kael)***
"Uma explosão de Nintendo." ***(Kojima BR)***
"A diversão da Nintendo com um BLAST de conteúdo." ***(Gustavobbp)***
"O Seu Portal de Noticias e Entretenimento." ***(Linkzx)***
"Uma explosão de informações nintendisticas para você." ***(Tico)***
"A cada jogo, uma nova emoção." ***(Crazy Dark Sonic)***

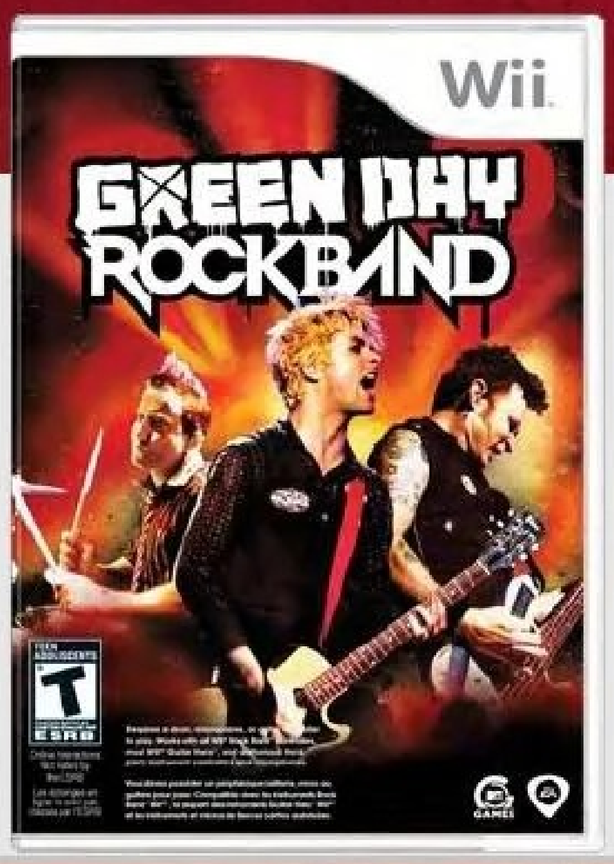
Parabéns aos selecionados, e até o próximo Custom Blast!

Próxima

por **Filipe Gatti**

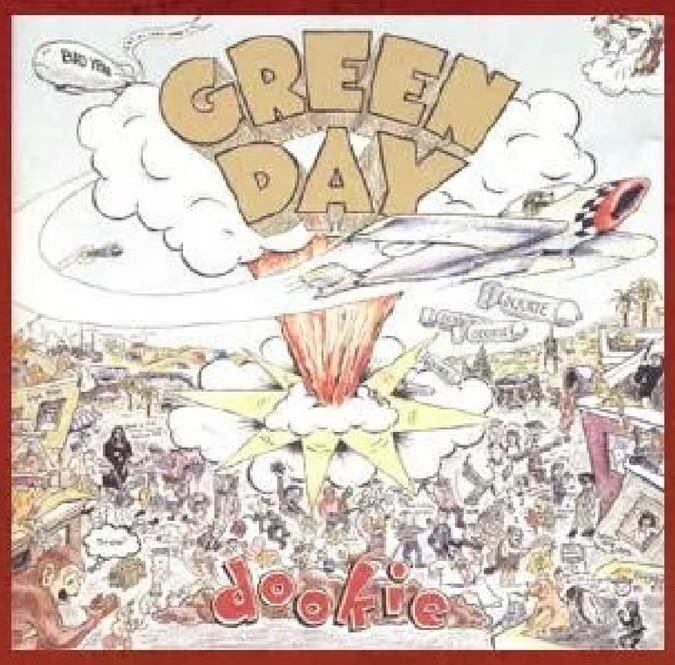
GREEN DAY ROCK BAND

Nada mais justo do que ser sincero logo de cara: se você não é fã da banda que habita o título do jogo, então não é necessário dar muito valor à este lançamento. A Harmonix, em conjunto da publisher MTV Games, lançam Green Day: Rock Band, título que não inova em nenhum parâmetro a conhecida franquia musical, mas oferece 47 músicas, 3 CDs inteiros, e uma boa gama de conteúdo extra e exclusivo que pode agradar bastante aos fãs da banda Californiana.



Green Day: Rock Band é um título indispensável para quem é fã da banda e gosta de jogar Rock Band com os amigos, afinal, são 47 músicas, entre vários extras. Mas se você não acha o gênero musical do grupo muito interessante – ou pior ainda – se você não suporta Green Day, pegue a nota abaixo e diminua 8,5 pontos

Os modos, os locais e suas histórias



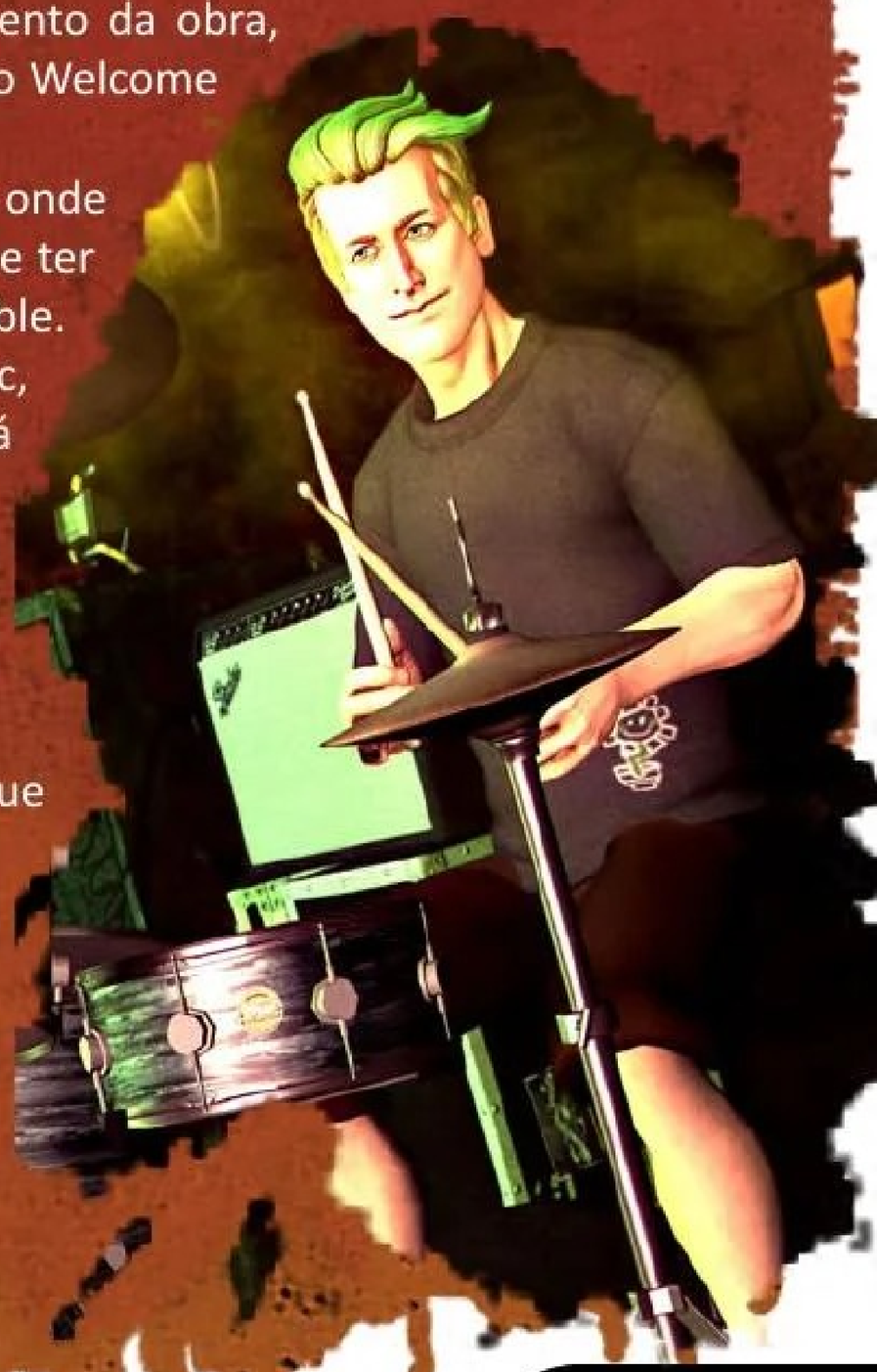
São dois modos de jogo principais neste Rock Band. O QuickPlay traz imediatamente todas as músicas habilitadas por padrão – um detalhe interessante, evitando que o jogador casual tenha que completar todo o modo carreira apenas para jogar as músicas principais com seus amigos. Já o citado Career oferece três localizações diferentes para tocar as músicas, cada uma com sua respectiva importância, característica, e faixas:

A The Warehouse, como descrito pelo próprio baixista Mike Dirnt, é um local fictício que representa bem o tipo de lugar em que a banda tocava no seu início de carreira. Nessa área são apresentadas as músicas do terceiro e provavelmente o mais aclamado CD do grupo, Dookie. O álbum inteiro está presente, então se você é fã da banda desde 1994, data de lançamento da obra, prepare-se para se emocionar enquanto toca sucessos como Welcome to Paradise, Basket Case, She e When I Come Around.

Em Milton Keynes, segundo local apresentado no modo, foi onde a banda fez o primeiro show em estádio de Londres, além de ter sido o cenário para a gravação do DVD ao vivo de Bullet in a Bible. Aqui as músicas variam entre os álbuns Nimrod, Insomniac, Warning e American Idiot, sendo que o último também está representado em sua totalidade.

Por fim, você pode jogar o CD inteiro de 21st Century Breakdown no Fox Theatre, em Oakland, que além de ser a cidade natal da banda, também foi o local onde eles fizeram um show de reabertura após 30 anos de reforma.

São apenas estas três localizações e que já vêm habilitadas por padrão, o que pode ser um ponto fraco para alguns. O interessante é a forma como cada uma é ambientada, seguindo fielmente as diferentes fases do grupo. Exemplo prático é ver que na The Warehouse o visual dos integrantes ainda é de uma típica banda de punk rock no seu começo de carreira, e no Fox Theatre todo o figurino se altera para um tom mais dark e melódico, que foi o rumo que o grupo tomou numa determinada fase.



O progresso do modo carreira e as novidades

É impossível não comparar Green Day: Rock Band com a versão de The Beatles da série, afinal, ambos compartilham o mesmo conceito. Acredito que a grande diferença entre os dois (tirando as músicas, óbvio) está no progresso do modo carreira. Enquanto o The Beatles apresentava cenas de corte, curiosidades, entre outros detalhes que representavam de alguma forma o seu progresso, Green Day: Rock Band não consegue simular isso de forma alguma.

Você simplesmente vai jogando as músicas nas três localizações existentes sem nenhum senso de progresso, afinal, até todas as músicas já estão habilitadas por padrão no QuickPlay.
Com o tempo, o que acontece é a possibilidade de habilitar "desafios" que nada mais são do que tocar uma setlist com quase ou total perfeição. Ao concluí-los, você recebe alguns extras como fotos raras, vídeos de shows, e até mesmo entrevistas exclusivas – o tipo de conteúdo que leva os fãs à loucura, e fazem Green Day: Rock Band um título que vale a pena para estes.

Sobre as novidades... bem, na verdade isso é tudo. O título não inova de nenhuma forma a mecânica de jogo. Os modos são os mesmos de outros jogos da série. Existem apenas três localizações diferentes, e o modo carreira não apresenta nenhum tipo de senso de progressão – mas que é recompensado com as dezenas de extras.

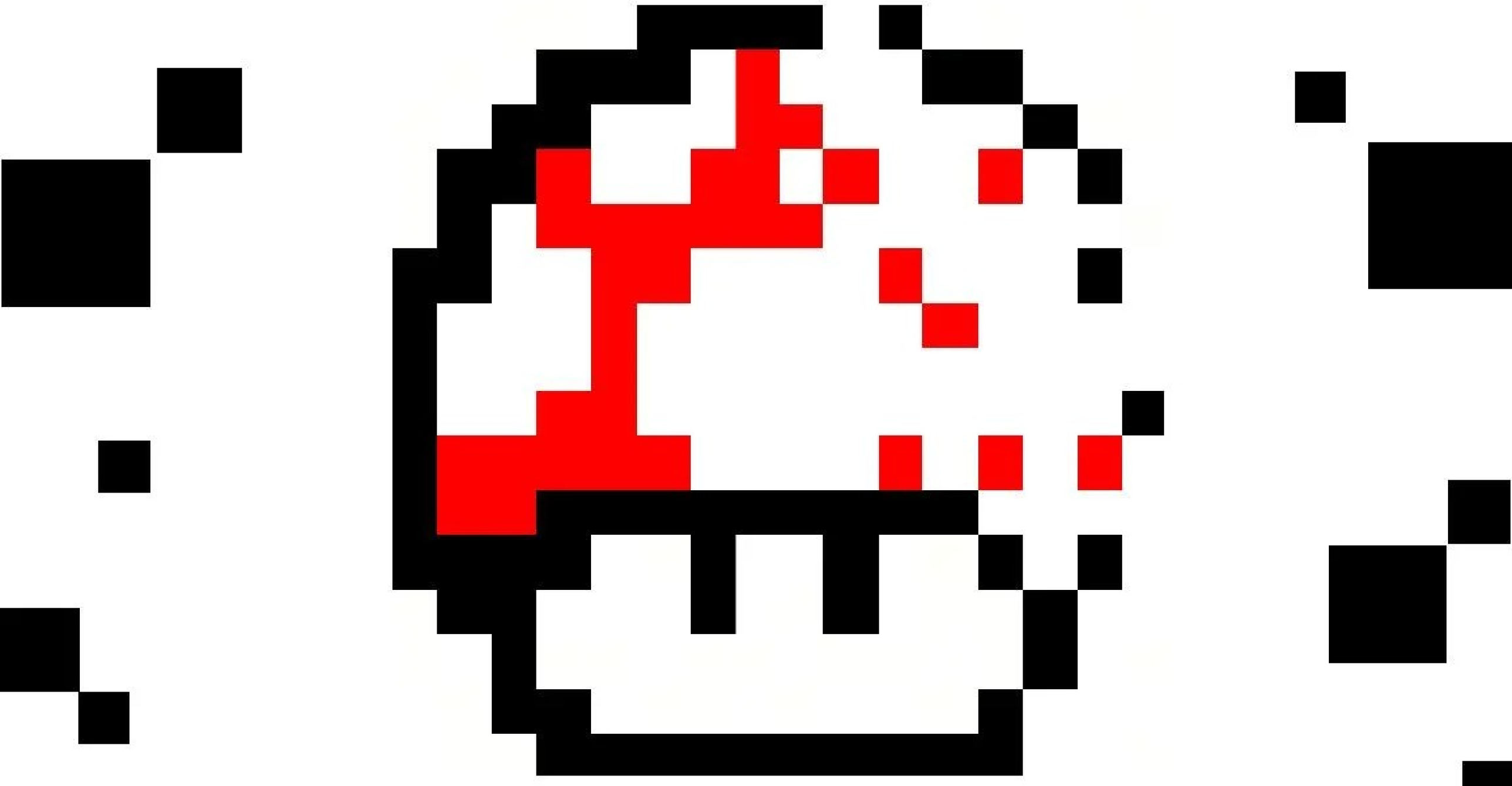
Gráficos e comentários finais

A ambientação, como citada anteriormente, está muito bem detalhada nos cenários, e também nos integrantes das bandas. Se você é um fã ávido, vai perceber os mais singelos detalhes de mudanças em cada local além da forma de se vestirem, como penteados e tatuagens que surgiram com o tempo.
O pacote gráfico, como um todo, segue o padrão de qualidade da Harmonix, então podemos esperar personagens bem modelados e animações sem queda nas taxas de quadros.

8.5

Green Day: Rock Band - (Wii)

GRÁFICOS **9,0** | SOM **9,0**
JOGABILIDADE **8.0** | DIVERSÃO **8,0**



GameDev

PARTE 8:

Criando, desenhando e trabalhando com sprites no XNA

Por **Sérgio Oliveira**

No nosso último encontro aqui na coluna GameDev criamos o nosso primeiro projeto de jogo XNA no Visual C# Express Edition, analisamos cada arquivo desse projeto e trouxemos à prática cada um dos conceitos vistos nas colunas anteriores. Ainda conseguimos visualizar cada um dos principais métodos do jogo e para que cada um deles serve.

No encontro de hoje, continuaremos a trabalhar no projeto vazio criado anteriormente. Entenderemos o que são sprites, os criaremos e aprenderemos a trabalhar com eles no nosso jogo – desenhando e fazendo-os movimentar na tela.

Se você não acompanhou o nosso último encontro e não tiver o projeto vazio que foi criado naquela oportunidade, poderá fazer o download dele acessando o endereço http://bit.ly/c1eKHw

Lembrando que, para desenvolvermos o que é proposto nessa coluna, é necessário ter instalado em sua máquina o XNA Game Studio e o Microsoft Visual C# Express Edition, ambos gratuitos, que podem ser encontrados na sessão downloads do XNA Creators Club (http://creators.xna.com/en-US/downloads).

Entendendo os termos e os conceitos dos gráficos 2D

Como hoje iremos trabalhar com gráficos 2D no nosso projeto, acredito que seja importante conhecermos um pouco dos termos que são utilizados no meio e que também utilizaremos daqui em diante.

- **Sprites:** é uma imagem 2D que pode ser manipulada independentemente do restante dos gráficos que estão na tela do jogo. Geralmente é o personagem que você controla nos jogos de plataforma, como o Mario em Super Mario World. Porém, ao contrário das demais imagens utilizadas no jogo, possui características definidas em nível de programação, como velocidade, posição, altura e largura. É importante perceber que o computador sempre irá desenhar as imagens como retângulos no jogo e é muito comum nós desenharmos imagens com áreas transparentes para dar noção de desenho não retangular. Ainda fala-se em sprite animado para denominar um sprite que muda de forma a cada determinado

- **Texturas**: é uma imagem 2D que é carregada sobre um modelo 3D para criar a ilusão de gráficos bem detalhados.

- **Billboard:** é um tipo especial de textura utilizada para representar objetos complexos sem que haja a necessidade renderizar um modelo 3D completo. Sua ideia consiste em, aonde pudesse haver um objeto 3D, haverá um sprite 2D com uma imagem da textura do objeto sobre ele. Obviamente não é uma técnica que substitui um modelo 3D e, por isso, é utilizada somente quando se deseja manter a performance de renderização e processamento de uma cena cujos detalhes devem

Confira um exemplo de Billboard no endereço http://creators.xna.com/en-US/sample/billboard

- **Background:** essa todos nós conhecemos – é a imagem que fica no plano de fundo nos jogos de plataforma 2D. Fala-se em 3 tipos de backgrounds: estáticos, animados e animados por paralaxe. Como o próprio nome diz, os estáticos não saem do canto e são background para uma tela apenas – Ninja Gaiden era desse jeito. Os backgrounds animados estão presentes nos jogos batizados de scrolling, como SMW. Já os backgrounds animados por paralaxe são formados por mais de um background, cada um deles se movendo a velocidades diferentes, e às vezes em sentidos opostos, dando a falsa impressão de tridimensionalidade e profundidade de cenário.

Não tem como fugir desses termos quando trabalhamos com jogos em 2D. No entanto, como iremos trabalhar apenas os conceitos de desenhar, movimentar e fazer colidir dois objetos no jogo, trabalharemos apenas o conceito de sprites.

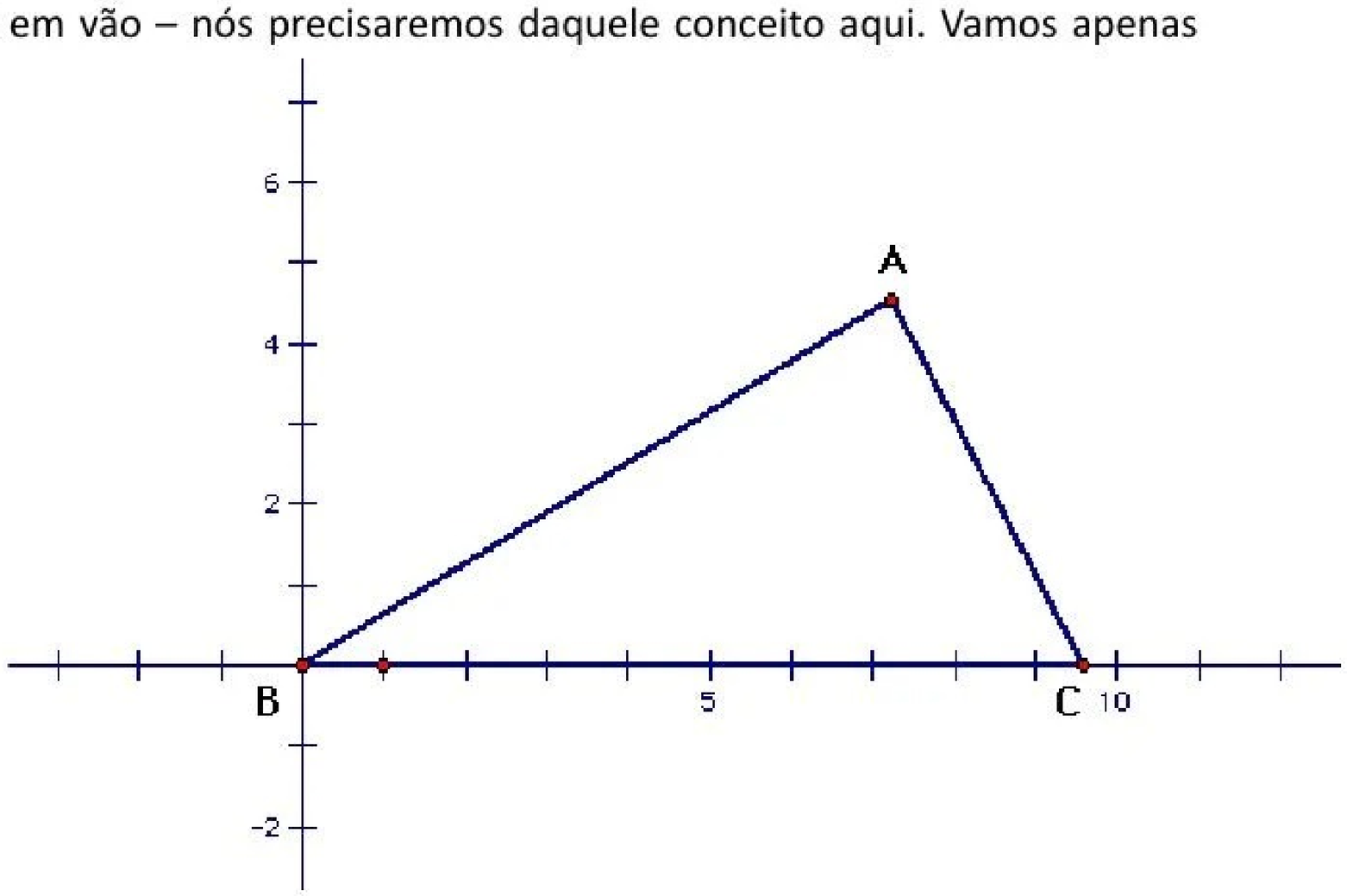
Porém, antes de sairmos criando os nossos sprites e fazermos eles colidirem um com o outro, também é importante ter uma noção mínima de como funciona o sistema de coordenadas 2D no XNA, se não será impossível compreendermos o que estamos fazendo e aonde estamos fazendo.

2D e o sistema de coordenadas da tela

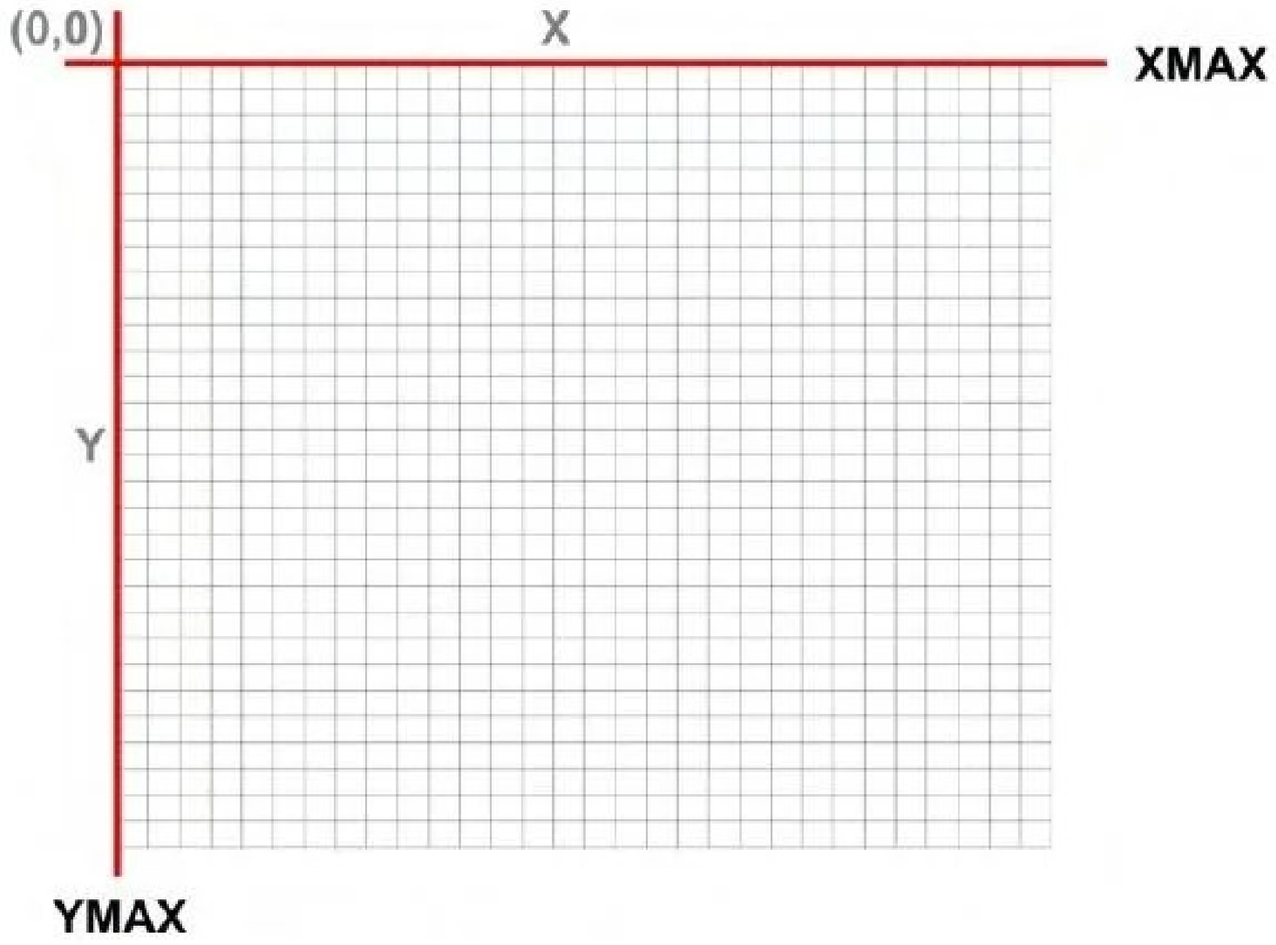
Lembra-se daquelas aulas chatas de geometria plana que você teve no seu ensino fundamental e até mesmo no ensino médio? Quem nunca quis esganar o professor por causa daqueles inúmeros planos cartesianos desenhados no caderno para calcular a distância entre dois pontos, área do triângulo, e tudo o mais a partir das coordenadas cartesianas, que atire o primeiro esquadro.

Pois bem, saiba que aquilo tudo não foi em vão – nós precisaremos daquele conceito aqui. Vamos apenas recapitular como funcionam as coordenadas 2D para que possamos compreender a maneira correta de posicionar os nossos objetos dentro do jogo.

Lá nas aulas de geometria nós utilizávamos o plano cartesiano comum para definirmos as figuras geométricas:



Perceba que nesse plano a origem dos pontos se dá no canto inferior esquerdo e o valor de Y é crescente quando o seu eixo sobe, e decrescente quando desce. Quando estamos trabalhando com computadores, no entanto, também estamos trabalhando com o sistema de coordenadas de tela, que é semelhante ao cartesiano:



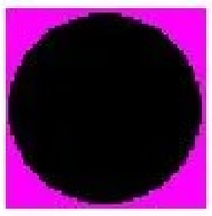
Analisando a figura, podemos perceber as três diferenças existentes entre esses dois sistemas. Primeiro, a origem dos eixos se dá na parte superior esquerda do plano. Segundo, o valor de Y é crescente quando seu eixo cresce para baixo e, por último, os valores de Y e X terão um valor máximo que não deverá ser extrapolado.

Ou seja, se você estiver trabalhando em uma tela cuja resolução seja 1024x768, o valor máximo que X poderá assumir é 1024 e Y é 768.

Desenhando um sprite no XNA

Agora sim, vamos começar a usar o XNA para desenhar um sprite na tela do nosso jogo. Para efeitos de demonstração, utilizaremos no encontro de hoje uma imagem simples que pode ser criada no próprio Microsoft Paint – ou você pode pegar qualquer imagem pronta por aí.

Aqui, criei uma imagem de 64x64 pixels, com fundo magenta e que atribui o nome de bola.bmp. Salvei-a em BMP para que eu pudesse esconder a área em magenta, deixá-la transparente. Se eu salvasse-a em JPG, isso não seria possível, pois esse formato não preserva o formato original das cores quando salvamos a imagem.



Agora que temos a imagem criada, devemos ir até o Visual C# Express Edition para criar a classe para os nossos sprites. A classe será criada para agruparmos a imagem do sprite e associá-la a algumas propriedades – tais como tamanho, posição e velocidade. Por enquanto essa classe será bastante simples e iremos trabalhando-a ao longo do desenvolvimento dos nossos encontros.

Observação: como eu havia dito no encontro anterior, nós faremos uso de alguns termos que são comuns a quem trabalha com programação orientada a objetos e que podem ser totalmente novas para aqueles que nunca nem ouviram falar em programação. A ideia e conceito de classe é um desses termos e para que ninguém fique perdido, novamente sugiro a leitura de uma apostila de introdução a programação orientada a objetos e/ou artigos da wikipedia que abordem o assunto. Para entender um pouco mais sobre classes, acessem o link http://pt.wikipedia.org/wiki/Classe_(programação)

Para criar uma classe no projeto, clique com botão direito em cima do nome do projeto no Solution Explorer do Visual C# Express Edition e selecione a opção Add/New Item (Figura 1) e logo em seguida escolha Class e atribua um nome à sua classe – aqui eu atribui o nome classSprite.cs (Figura 2)

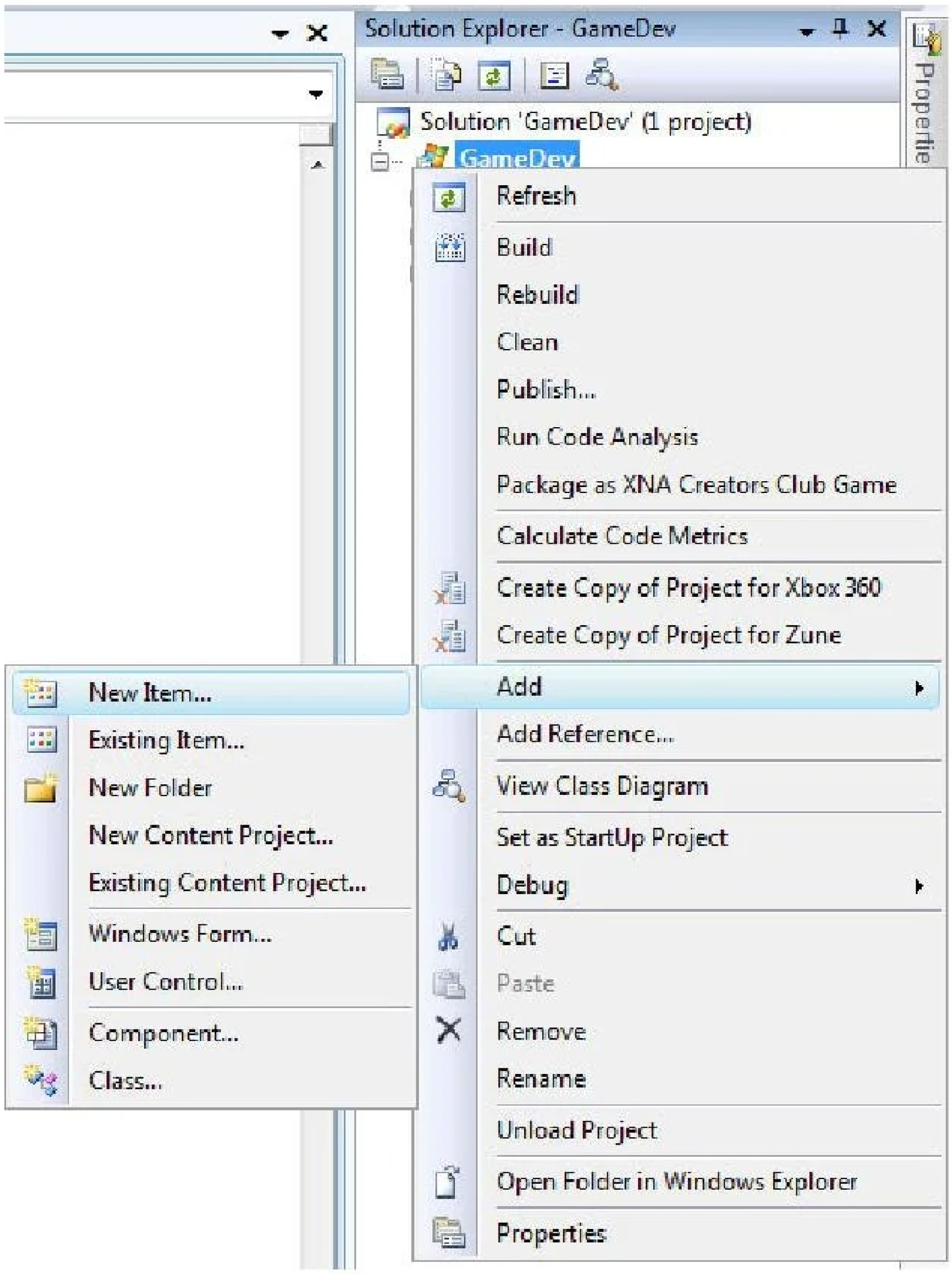


Figura 1

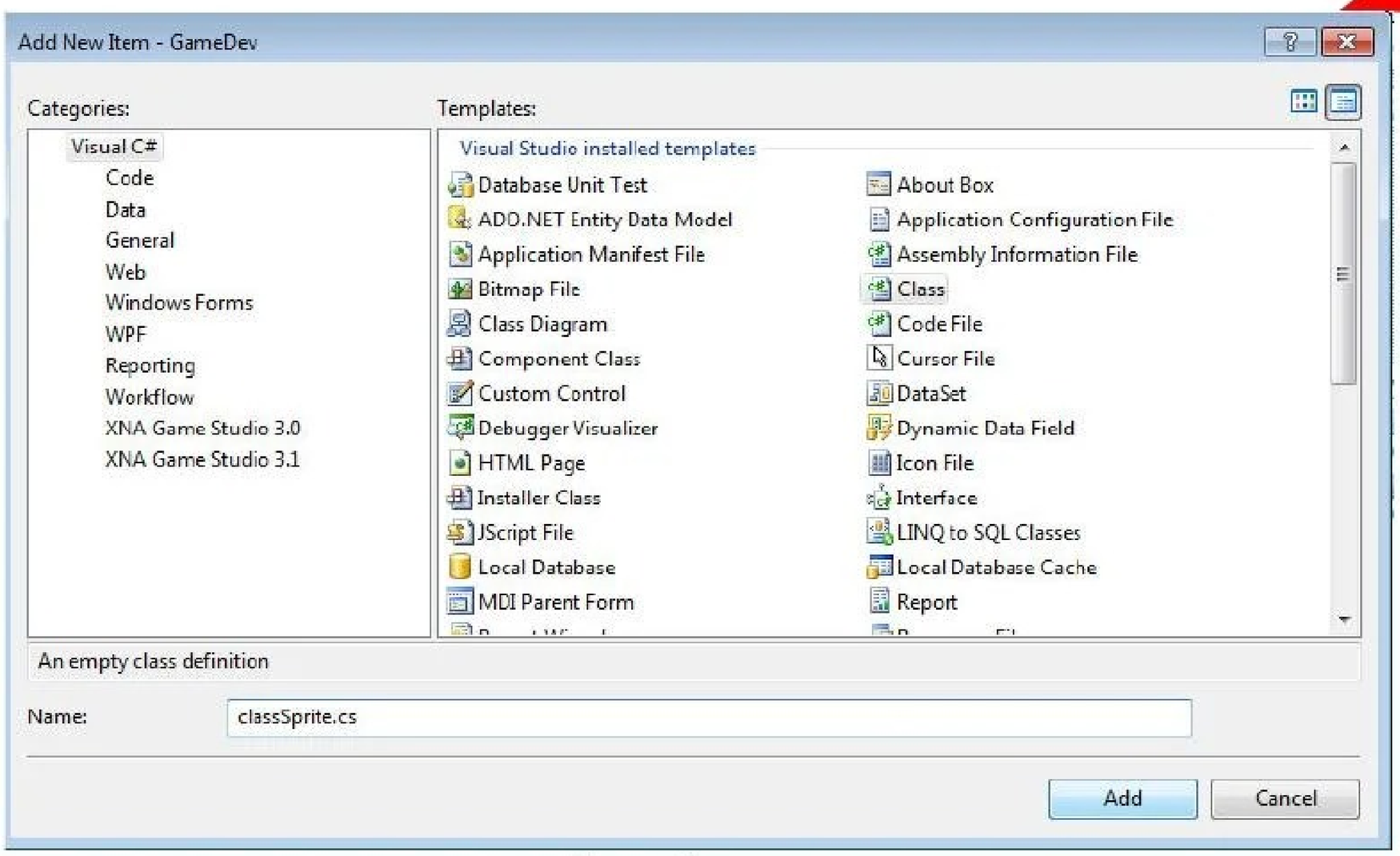


Figura 2

Ao finalizarmos a criação da classe, o Visual C# nos levará automaticamente para ela e nos revelará uma estrutura básica que foi gerada automaticamente. Vamos trabalhar um pouco nessa estrutura – vamos apagar tudo o que está lá e adicionarmos as linhas que seguem abaixo no seu escopo:

```
using Microsoft.Xna.Framework.Graphics; //Para trabalharmos com texturas 2D
using Microsoft.Xna.Framework; //Para trabalharmos com vetores bidimensionais

class classSprite //Declaração padrão do escopo da classe
{
   public Texture2D textura { get; set; } //Declaração de uma textura 2D padrão
   public Vector2 posicao { get; set; } //Declaração de uma variável Vector2 chamada posicao para trabalharmos com a posição do sprite na tela
   public Vector2 tamanho { get; set; } //Declaração de uma variável Vector2 para definirmos o tamanho do sprite em pixels

   public classSprite(Texture2D novaTextura, Vector2 novaPosicao, Vector2 novoTamanho) //Declaração de um método construtor - para saber mais, visite: http://bit.ly/dcQZ5L
   {
      textura = novaTextura;
      posicao = novaPosicao;
      tamanho = novoTamanho;
   }
}
```

Por enquanto a nossa classe não tem nada de especial, mas aqui cabe explicar um pouco as três propriedades que estamos utilizando:

- **Textura:** armazenará a imagem do sprite usando a classe Texture2D do XNA. Essa classe nos fornece vários métodos que nos ajudarão a lidar com os sprites que utilizaremos.

- **Tamanho:** armazenará o tamanho do sprite utilizando a classe Vector2 do XNA. Essa classe tem duas propriedades – X e Y – que serão utilizadas, respectivamente, para definir a largura e altura do sprite.

- **Posição:** tal qual o tamanho, utilizará a classe Vector2 do XNA. Todavia, as propriedades X e Y armazenarão as coordenadas de tela para o posicionamento do sprite.

Agora que já criamos a imagem do nosso sprite e a classe dos sprites do nosso jogo, é hora de importar essa imagem para o **Content Pipeline** do projeto. Para fazer isso, basta clicar com o botão direito na pasta **Content** no *Solution Explorer* do projeto e selecionar **Add/Existing Item** (Figura 3). Na janela que abrir, procure pelo local em que você salvou a imagem do sprite e aperte em Add e voilá – a imagem do sprite estará lá na pasta Content do seu projeto (Figura 4)

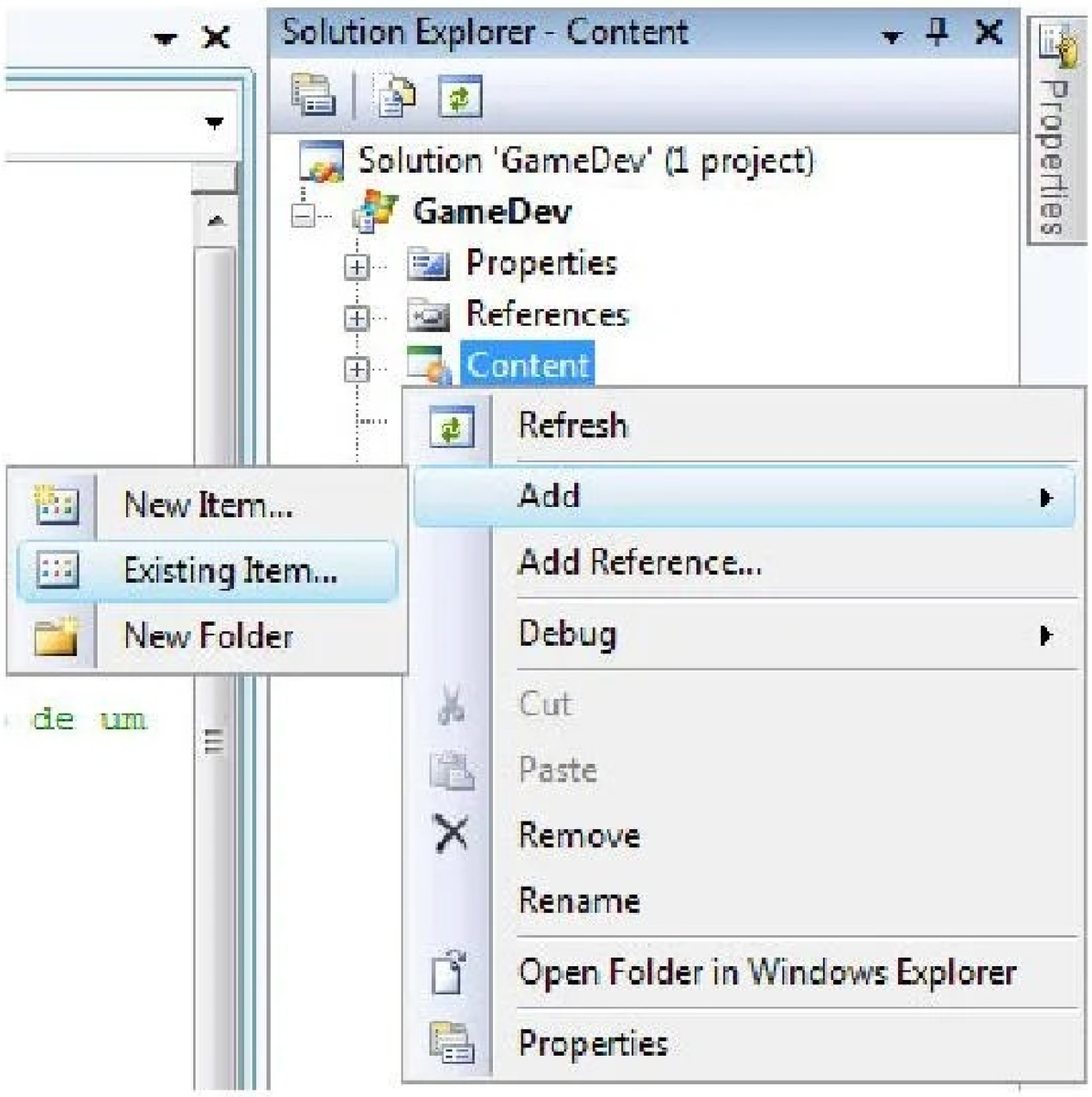


Figura 3

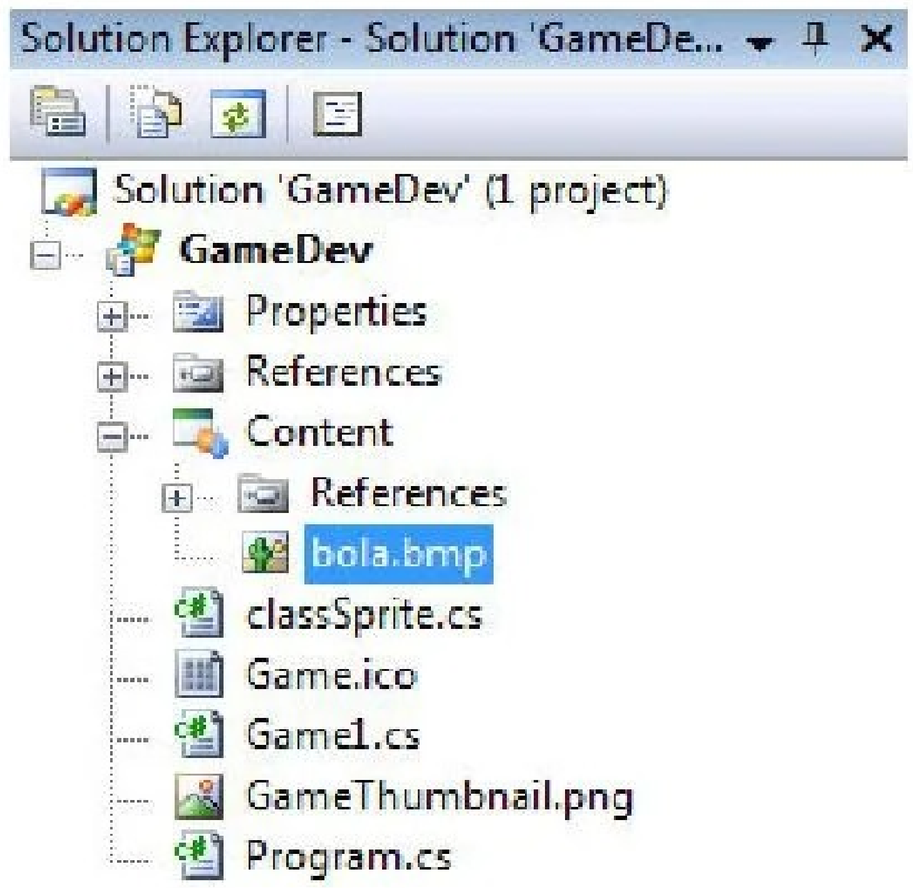


Figura 4

Agora que já temos a imagem do *sprite* criada, temos a classe do *sprite* no projeto, e a imagem devidamente importada para o *Content Pipeline*, nos resta desenhá-la na tela. Para tanto, nós utilizaremos a *SpriteBatch* - uma classe do XNA que auxilia o desenho de sprites na tela – e a textura/imagem que acabamos de carregar – nesse caso, essa textura será carregada para dentro da classe classSprite e então a utilizaremos.

Existem diversos meios que nos permitem desenhar o sprite na tela. Porém, geralmente ou se lê a textura na classe classSprite e desenha no método Draw da classe Game1, ou se adiciona um método Draw dentro da própria classe classSprite que desenhará, a partir da classe, o sprite na tela. Vamos trabalhar dessa forma por enquanto.

Voltando à nossa classe classSprite, adicionemos o seguinte método após o método construtor da classe:

```
public void Draw(SpriteBatch spriteBatch) //Método para desenhar o sprite na tela
{
  spriteBatch.Draw(textura, posicao, Color.White);
}
```

Aqui nós estamos usando o método Draw da classe SpriteBatch e estamos passando três argumentos para esse método – a textura, a posição e a cor. Essa é a maneira mais simples de desenhar algo na tela.

Caso queira conhecer outras formas de utilizar o método Draw, basta acessar o endereço http://bit.ly/cWHXPt

Agora precisamos ajustar a classe Game1. Ao acessarmos o arquivo, perceberemos que, no ato da criação do projeto, o Visual C# já criou um objeto spriteBatch no começo do arquivo. Vamos agora criar um objeto classSprite logo após a definição dos objetos graphics e spriteBatch. Teremos algo mais ou menos assim:

```
public class Game1 : Microsoft.Xna.Framework.Game
{
  GraphicsDeviceManager graphics; //O dispositivo de vídeo
  SpriteBatch spriteBatch; //O renderizador de sprites

  classSprite mySprite1; //Minha classe de sprites
```

Como tinhamos comentado no nosso encontro anterior, é preciso carregar o conteúdo que utilizaremos no nosso jogo já no método **LoadContent,** que possui toda a inicialização gráfica do jogo. Então, vamos ao método LoadContent da classe *Game1* e inicializemos o objeto **mySprite1:**

```
protected override void LoadContent()
{
      //Cria um sprite com textura 2D
      mySprite1 = new classSprite(Content.Load<Texture2D>("bola"),
                  new Vector2(0f, 0f), new Vector2(64f, 64f));

      // Create a new SpriteBatch, which can be used to draw textures.
      spriteBatch = new SpriteBatch(GraphicsDevice);
}
```

É aqui nessa parte que o método construtor da classe *classSprite* se mostra necessário – a partir da inicialização de *mySprite1*, já podemos definir que "aspecto" ele terá. Carregamos nele a textura que importamos há pouco tempo (bola.bmp), a posição inicial do nosso sprite (0, 0 – ou seja, o canto superior esquerdo da tela) e o tamanho do sprite (64, 64). Apesar das poucas linhas de código, até aqui já fizemos muitas coisas.

Uma dica de boa prática de programação é destruir tudo aquilo que foi criado por você após a finalização do jogo. Para quem acompanha a coluna, já deve saber que isso deverá ser feito no método UnloadContent. Adicionemos a seguinte linha a ele:

```
protected override void UnloadContent()
{
        mySprite1.textura.Dispose();
}
```

Finalmente, podemos ir ao método Draw da classe Game1 para desenhar o nosso sprite na tela:

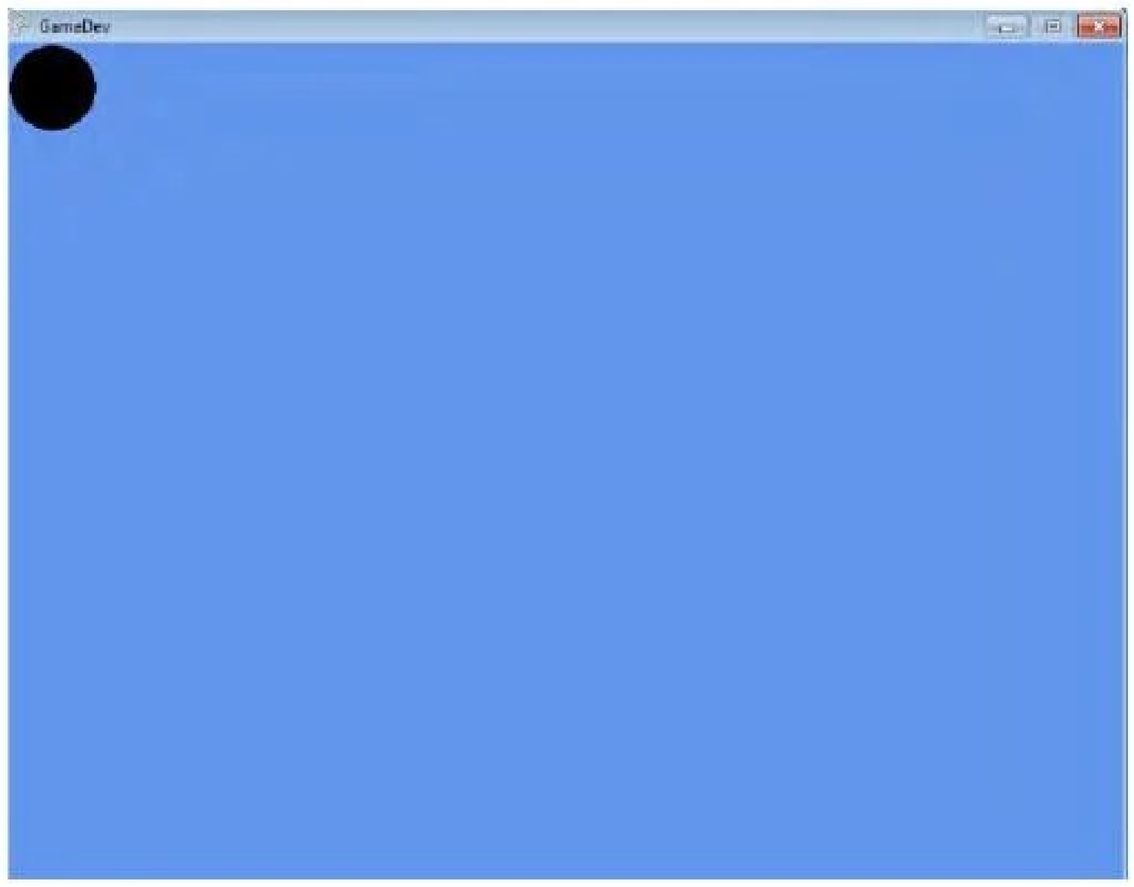
```
protected override void Draw(GameTime gameTime)
{
        GraphicsDevice.Clear(Color.CornflowerBlue);

        spriteBatch.Begin();
        mySprite1.Draw(spriteBatch);
        spriteBatch.End();

        base.Draw(gameTime);
}
```

Aqui, estamos simplesmente dizendo ao XNA que começaremos a desenhar algo na tela (*spriteBatch.Begin*). Logo em seguida, desenhamos o que desejamos (*mySprite1.Draw*) e finalmente dizemos ao XNA que tudo foi desenhado (*spriteBatch.End*). Fácil, não?

Uma dica importante aqui – como estamos trabalhando com uma textura que possui áreas de transparência (lembra para que serve o magenta do fundo da nossa bola?), precisamos dizer ao *spriteBatch* isso. Para tanto, basta modificar o *spriteBatch.Begin* para:

```
spriteBatch.Begin(SpriteBlendMode.AlphaBlend);
```

Estamos com o circo armado – se você seguiu todos os passos atentamente, ao dar F5 e mandar o "jogo" rodar, obterá a seguinte tela:



Perfeito – antes tinhamos uma grande tela azul, agora temos uma grande tela azul com um ponto preto no canto superior esquerdo. Você pode brincar com o posicionamento da bola lá no método *LoadContent*, basta alterar os valores que lá se encontram – se divirta, mexa e procure entender como tudo está funcionando mexendo por si próprio antes de seguir para o próximo passo.

Movendo o sprite na tela

Como estamos trabalhando com gráficos 2D nesse projeto, é fácil pensar em como iremos fazer com que nosso sprite se mova na tela. Alguém já consegue visualizar como faremos isso?
Na verdade não tem nada de complicado aqui – é tudo bem lógico. Para movermos o nosso sprite para a direita, basta incrementar a coordenada X da sua posição. Para movê-lo para a esquerda, basta decrementar o valor da coordenada X. O mesmo vale para Y – se quiser que desça, incremente seu valor; se quiser que suba, decremente-o.

É importante não esquecer como funciona o sistema de coordenadas da tela do computador nesse momento.

Na estrutura básica do jogo, o XNA nos fornece um método específico para realizarmos cálculos durante a execução do jogo – é o **Update**. Aqui, poderemos adicionar as lógicas de cálculo de posicionamento, velocidade e muito mais.

Para que o sprite possa mover na tela, podemos adicionar uma única linha de código dentro do método *Update*:

```
//Mudará a posição X do sprite a cada loop do gameTime
mySprite1.posicao.X += 1;
```

Aperte F5 e observe que a bola se moverá para a direita infinitamente, até desaparecer da tela. Tá, funciona, mas está feio. Vamos fazer algo mais elegante e bonito. Vamos fazer com que o nosso *sprite* fique “preso” aos limites da tela do jogo.

Para isso, primeiramente apague a linha que digitou acima no método *Update* da classe *Game1*. Logo em seguida, vamos à classe classSprite para fazermos algumas pequenas modificações.

O que iremos fazer é declarar uma nova propriedade chamada “velocidade” que indicará a velocidade do sprite nas coordenadas X e Y da tela. Também definiremos uma propriedade que armazenará a resolução atual da tela do jogador para que possamos limitar a área de movimentação do nosso sprite.

Teremos, então, o seguinte:
Em classSprite.cs

```
private Vector2 tamanhoTela { get; set; } //Armazenará a resolução, o tamanho, atual da tela
public Vector2 velocidade { get; set; } //Indicará a velocidade do sprite em X e Y
```

```
public classSprite(Texture2D novaTextura, Vector2 novaPosicao, Vector2 novoTamanho, int larguraTela, int
alturaTela) //Declaração de um método construtor - para saber mais, visite: http://bit.ly/dcQZ5L
{
        textura = novaTextura;posicao = novaPosicao;
        tamanho = novoTamanho;
        tamanhoTela = new Vector2(larguraTela, alturaTela);
}
```

Agora voltemos à classe *Game1* para alterarmos a inicialização de *mySprite1* no método *LoadContent*:

```
//Cria um sprite com textura 2D
mySprite1 = new classSprite(Content.Load<Texture2D>("bola"),
          new Vector2(0f, 0f), new Vector2(64f, 64f),
          graphics.PreferredBackBufferWidth, graphics.PreferredBackBufferHeight);
mySprite1.velocidade = new Vector2(1, 1); //Definindo a velocidade do sprite.
```

Como havíamos definido na classe do nosso *sprite* que agora armazenaríamos a resolução da tela, aqui na instanciação do objeto do *sprite* estamos capturando a largura e altura da tela do jogo com o comando *graphics.PreferredBackBufferWidth* e *Height*.

Em seguida, estamos definindo a velocidade do nosso sprite. Ao dizermos que ele terá velocidade 1 para horizontal e 1 para vertical, estamos dizendo ao XNA que o nosso objeto se moverá 1 pixel por atualização de tela, tanto na coordenada X, quanto Y – ou seja, ele se moverá na diagonal.

Nós já temos a velocidade do nosso sprite e já sabemos quais os limites de tela e, consequentemente, de movimentação do sprite – agora só está faltando um método que fará com que o sprite se mova dentro dos limites da tela.

Na classe *classSprite*, abaixo do método *Draw*, adicionemos um método, que chamarei aqui de "Mover", que fará o sprite se mover dentro dos limites da tela.

```
public void Mover()
{
   //Se o sprite atingir um dos limites da tela, inverta a sua velocidade
   //Verificando o limite da direita da tela
   if (posicao.X + tamanho.X + velocidade.X > tamanhoTela.X)
      velocidade = new Vector2(-velocidade.X, velocidade.Y);

   //Verificando o limite inferior
   if (posicao.Y + tamanho.Y + velocidade.Y > tamanhoTela.Y)
      velocidade = new Vector2(velocidade.X, -velocidade.Y);

   //Verificando o limite da esquerda da tela
   if (posicao.X + velocidade.X < 0)
      velocidade = new Vector2(-velocidade.X, velocidade.Y);
```

```
    //Verificando o limite superior
    if (posicao.Y + velocidade.Y < 0)
      velocidade = new Vector2(velocidade.X, - velocidade.Y);

    //Uma vez que ajustamos a velocidade, adicione-a a posição atual do sprite
    posicao += velocidade;
}
```

Calma, não precisa se desesperar com o tamanho do código e com esses "if". Basicamente, o que estamos fazendo aqui é verificar se o *sprite* já atingiu a posição limite da tela e, caso tenha atingido, terá sua velocidade invertida.

Ora, é bem lógico. Como havia explicado anteriormente, para mover o *sprite* para a direita, basta incrementar o valor de X – para mover para a esquerda, basta decrementar. Então, se o *sprite* alcançou o limite da direita da tela, o que temos que fazer? Decrementar o valor de sua velocidade no eixo X, pois é a velocidade que fará com que o sprite mude de posição com o passar do tempo – correto?

Dessa forma, só nos resta um passo para finalizarmos essa etapa – adicionar uma chamada do método "Mover" no *Update* de *Game1*:

```
mySprite1.Mover(); //Moverá o sprite dentro dos limites da tela do jogo
```

Pronto, está feito. Ao apertar F5 você perceberá que o sprite se movimentará dentro dos limites da janela do jogo! Para consolidar o aprendizado, tente alterar a velocidade do sprite e busque por aí a fora uma forma de alterar o tamanho da tela do jogo. Não esqueça também de ler atentamente tudo o que foi feito para garantir que não houve nenhuma dúvida.

Hoje nós aprendemos a desenhar um sprite na tela, adicionamos movimentação a ele e ainda trabalhamos com os limites da tela. Também construímos a nossa primeira classe – a classSprite – para armazenar os sprites que utilizaremos no jogo daqui para frente. No nosso próximo encontro, continuaremos a utilizar o projeto de hoje e adicionaremos detectores de colisão para duas ou mais bolas no nosso jogo. Fique ligado, estude, tire suas dúvidas e até lá!

O projeto atualizado com o código desse encontro pode ser baixado no endereço http://bit.ly/9xrJJS

NINTENDO BLAST

Confira outras edições em:
revista.nintendoblast.com.br